UNIT - I

1

INFORMATION SECURITY OVERVIEW

Unit Structure

1.0 OBJECTIVES

- Security means protecting data and information. Computer security has four objectives: confidentiality, integrity, availability, and nonrepudiation.
- Securing information is ensuring that computers keep your secrets, hold valid information and keep records of all the user's transactions in a secure manner.

1.1 DEFINITION

- Information is an important asset.
- Information can be classified into different categories.

- This is typically done to control access to the information in different ways, depending on its importance, its sensitivity, and its vulnerability to theft or misuse.
- For eg: The U.S. government, uses a five-level classification system that progresses from Unclassified information (which everyone can see) to Top Secret information (to which only the most trusted people have access).

1.2 INTRODUCTION

- Organizations classify information in different ways to differently manage aspects of its handling, such as labeling (whether headers, footers, and watermarks specify how it should be handled), distribution (who gets to see it), duplication (how copies are made and handled), release (how it is provided to outsiders), storage (where it is kept), encryption (if required), disposal (whether it is shredded or strongly wiped), and methods of transmission (such as e-mail, fax, print, and mail).
- Information intended for internal use only is usually meant to be seen by employees, contractors, and service providers, but not by the general public.
- Examples include internal memos, correspondence, general e-mail and instant message discussions, company announcements, meeting requests, and general presentation materials.

Companies may have confidential information, such as research and development plans, manufacturing processes, strategic corporate information, product roadmaps, process descriptions, customer contact information, financial forecasts, and earnings announcements that are intended for internal use on a need-to-know basis. Loss or theft of confidential information could violate the privacy of individuals

Specialized information or secret information may include trade secrets, such as formulas, production details, and other intellectual property, proprietary methodologies and practices that describe how services are provided, research plans, electronic codes, passwords, and encryption keys. If disclosed, this type of information may severely damage the company's competitive advantage.

A Case Study:

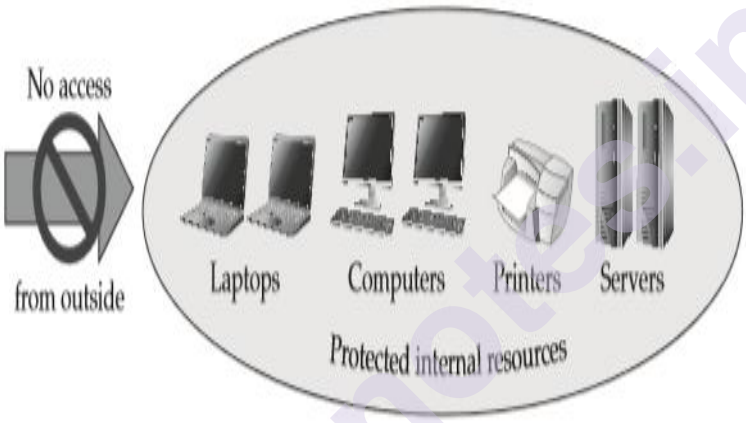
- Egghead Software was a well-known software retailer that discovered in 2000 that Internet attackers might have stolen as many as 3.7 million credit card numbers from its website, housed offsite at an e-commerce service provider that lacked good security.
- This information quickly made the news.
- The media coverage cleaned out the company's reputation. Egghead's stock price dropped dramatically, along with its sales.

- In some business sectors, the protection of information is not just desirable, it's mandatory. For example, healthcare organizations are heavily regulated.
- Regulations also required financial institutions to protect customer information, PII, and financial records.

1.3 THE EVOLUTION OF INFORMATION SECURITY

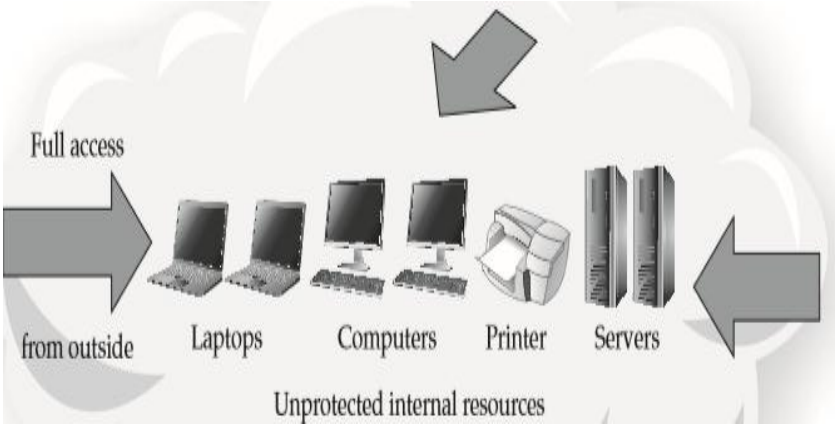
1.3.1 Government perimeter blockade model:

- The government was mainly concerned with blocking access to computers, restricting internal access to confidential data, and preventing interception of data.
- This method of protecting assets provided a hard-to-penetrate perimeter, as shown below.



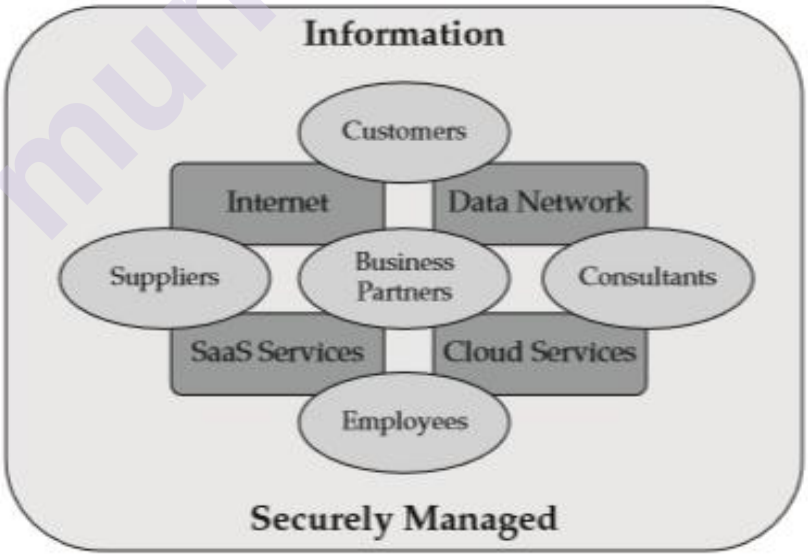
1.3.2 Academic world:

- The goal was to share information openly, so security controls were limited to accounting functions in order to charge money for the use of computer time.
- These two models are diametrically opposite—the government model blocks everything, while the academic model allows everything.



Case Study: Dangers of the Academic Open-Access Model:

- InterNex was an Internet service provider (ISP) headquartered in Palo Alto, California. The only security control was the basic username and password authentication.
- The ideology of InterNex was that the Internet should be open to everyone. Many of its systems were compromised by attackers who were able to guess the passwords of various user accounts.
- One of the most famous attackers, Kevin Mitnick, used InterNex's systems while attacking other networks, including during the 1994 IP spoofing attack against computers in San Diego. Mitnick was eventually captured and served five years in jail.
- The concepts of intranets and extranets were developed to accommodate internal and external customers, respectively, with secured boundaries that resembled miniature versions of the firewall perimeter.
- Virtual private networks (VPNs) were developed to provide a secure channel (or tunnel) from one network to another.
- As more companies started doing business on the Internet, concepts such as **Software-as-a-Service (SaaS)** were developed to provide business services over the Internet.
- And the threats found on the Internet evolved as well. Basic viruses and worms along with man-in-the-middle attacks found.



Modern information is shared among many consumers, via many channels.

Security Methodology:

- The field of security is concerned with protecting assets in general.

- Information security is concerned with protecting information in all its forms, whether written, spoken, electronic, graphical, or using other methods of communication.
- Network security is concerned with protecting data, hardware, and software on a computer network.

1.4 THREE D'S OF SECURITY

Defense, Detection, and Deterrence.

The three D's of security

Defensive:

- Its controls on the network can include access control devices such as stateful firewalls, network access control, spam and malware filtering, web content filtering, and change control processes.
- These controls protect from software vulnerabilities, bugs, attack scripts, ethical and policy violations, accidental data damage, and the like.

Detective:

- Its controls include video surveillance cameras in local stores, motion sensors, and house or car alarm systems that alert passers-by of an attempted violation of a security perimeter.
- Detective controls on the network include audit trails and log files, system and network intrusion detection and prevention systems, and security information and event management (SIEM) alerts, reports, and dashboards.
- A security operations center (SOC) can be used to monitor these controls. Without adequate detection, a security breach may go unnoticed for hours, days, or even forever.

Deterrence:

- It is another aspect of security. It is considered to be an effective method of reducing the frequency of security compromises, and thereby the total loss due to security incidents.

- Many companies implement deterrent controls for their employees, using threats of discipline and termination for violations of policy.
- These deterrent controls include communication programs to employees about acceptable usage and security policies, monitoring of web browsing behavior, training programs to acquaint employees with acceptable usage of company computer systems, and employee signatures on agreements indicating that they understand and will comply with security policies.

With the use of deterrent controls such as these, attackers may decide not to cause damage.

Case Study on the Illusion of Security:

- Many drivers of Toyota vehicles in the 1980s were unaware that the door keys for those vehicles had only a small number of variations.
- They naturally assumed that so many different keys existed, the chance of opening the door of the wrong car was practically impossible. They were wrong.
- Toyota had so few key variations that thieves were able to carry a full set to steal the cars.
- One person who encountered this phenomenon was Betty Vaughn, a retired schoolteacher, Betty returned from a shopping trip to the local mall to find her Toyota's passenger-side mirror broken off and the garage door opener missing.
- When her husband arrived home, he noticed the front license plate was also missing. They assumed their car had been vandalized. But the tires were the wrong brand! What kind of vandal would switch their tires?
- It was then that they checked and discovered from the registration that it wasn't their car.
- The 1992 Toyota Camry had been parked two cars away from Charles Lester's 1993 model. The keys to both vehicles were the same.

1.5 HOW TO BUILD A SECURITY PROGRAM?

There are many components that go into the building of a security program:

- **Authority:** The security program must include the right level of responsibility and authorization to be effective.
- **Framework:** A security framework provides a defensible approach to building the program.
- **Assessment:** Assessing what needs to be protected, why, and how leads to a strategy for improving the security posture.

- **Planning:** It produces priorities and timelines for security initiatives.
- **Action:** The actions of the security team produce the desired results based on the plans.
- **Maintenance:** The end stage of the parts of the security program that have reached maturity is to maintain them.

1.5.1 Authority:

- A security program charter defines the purpose, scope, and responsibilities of the security organization and gives formal authority to the program.
- Usually, the security organization is responsible for information protection, risk management, monitoring, and response.
- Other responsibilities may include physical security, disaster recovery and business continuity planning, regulatory and internal compliance, and auditing.
- The set of responsibilities varies by company but should be clearly specified in the security program charter, which should be authorized by the company's executive staff.

1.5.2 Framework:

- The security policy provides a framework for the security effort.
- The policy describes the intent of executive management concerning what must be done to comply with the business requirements.
- The policy drives all aspects of technical implementations, as well as policies and procedures. Ideally, a security policy should be documented and published before any implementation begins.
- The security policy represents business decisions about what to do based on certain assumptions.
- If the assumptions are not documented, they may be unclear or conflict with other activities.
- Documenting these assumptions in a clear, easy-to-read, accessible policy helps communicate expectations to everyone involved.
- Standards are the appropriate place for product-specific configurations to be detailed.
- Standards are documented to provide continuity and consistency in the implementation and management of network resources.

1.5.3 Assessment:

- Risk analysis provides a perspective on current risks to the organization's assets.
- This analysis is used to prioritize work efforts and budget allocation so that the greater risks can receive a greater share of attention and resources.
- A risk analysis results in a well-defined set of risks that the organization is concerned about. These risks can be mitigated, transferred, or accepted.
- A gap analysis compares the desired state of the security program with the actual current state and identifies the differences.
- Those differences, or gaps, form a collection of objectives to be acted on over the course of a remediation effort to improve the organization's security posture to bring it in line with one or more standards, requirements, or strategies.
- Remediation planning considers the risks, gaps, and other objectives of the security program, and puts them together into a prioritized set of steps to move the security program from where it is today to where it needs to be at a future point.

1.5.4 Planning:

- A roadmap is a plan of action for how to implement the security remediation plans. It describes when, where, and what is planned.
- The roadmap is useful for managers who need the information to plan activities and to target specific implementation dates and the order of actions.
- It is also useful for implementers who will be responsible for putting everything together.
- The roadmap is a relatively high-level document that contains information about major activities and milestones coming up in the next defined period (often some combination of quarters, one year, three years, five years, or a "rolling" period that advances periodically).
- A good tool for architecture documents is a block diagram— a diagram that shows the various components of a security architecture at a relatively high level so the reader can see how the components work together.
- A block diagram does not show individual network devices, machines, and peripherals, but it does show the primary building blocks of the architecture.

- Block diagrams describe how various components interact, but they don't necessarily specify who made those components, where to buy them, what commands to type in, and so on.

1.5.5 Action:

- This describes how processes are performed by people on an ongoing basis to produce the desired outcomes of the security program in a repeatable, reliable fashion.
- Maintenance and support are part of maintaining the ongoing operations of the security program and its associated technologies, as part of a normal lifecycle of planning, updating, reviewing, and improving.
- The actions that should be taken when a security event occurs are defined in the incident response plan.

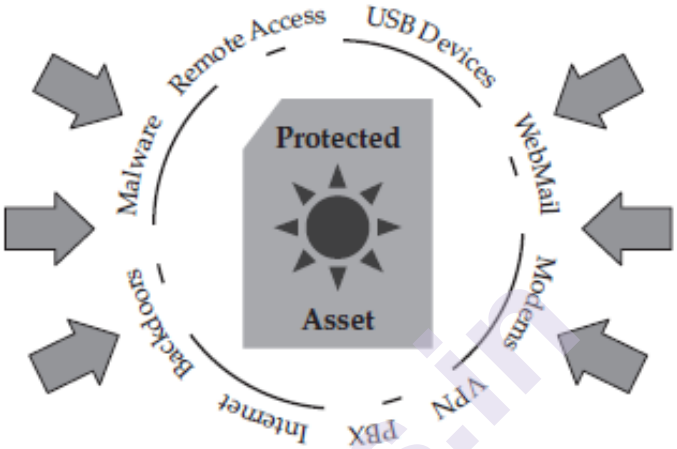
1.5.6 Maintenance:

- Policy enforcement is necessary to ensure that the intentions of management are carried out by the various people responsible for the behavior and actions defined in the security policies.
- Often, this enforcement is a shared effort between security management, company management, and Human Resources.
- Security awareness programs are used to educate employees, business partners, and other stakeholders about what behaviors are expected of them, what actions they should take under various circumstances to comply with security policies, and what consequences may ensue if they don't follow the rules.

1.6 THE IMPOSSIBLE JOB

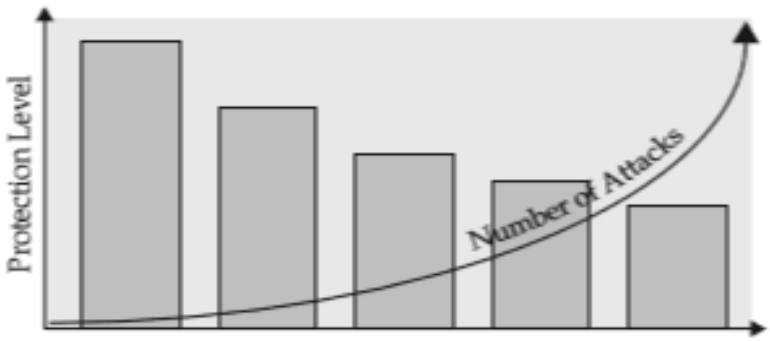
- The job of the attacker is always easier than the job of the defender.
- The attacker needs only to find one weakness, while the defender must try to cover all possible vulnerabilities.
- The attacker has no rules—the attacker can follow unusual paths, abuse the trust of the system, or resort to destructive practices.
- Defenders must try to keep their assets intact, minimize damage, and keep costs down.
- Eg: Homeowners who want to protect their property must try to anticipate every attack that is likely to happen, while attackers can simply use, bend, break, or mutilate the house's defenses.
- In an extreme example, the attacker can cut through the exterior, break the windows, knock down the walls, or set the house on fire.

- Homeowners have the more difficult job, of trying to protect their assets against all types of attacks.
- Every defender performs a risk assessment by choosing which threats to defend against, which to insure against, and which to ignore.
- Mitigation is the process of defense, transference is the process of insurance, and acceptance is deciding that the risk does not require any action.



1.7 THE WEAKEST LINK

- A security infrastructure will drive an attacker to the weakest link.
- For example, a potential intruder who is trying to break into a house may start with the front door. If the front door lock is too difficult to pick, the intruder may try side doors, back doors, and other entrances.
- If the intruder can't get through any of those, he may try to open a window. If they're all locked, he may try to break one. If the windows are unbreakable or barred, he may try to find other weaknesses.
- If the doors, windows, roof, and basement are all impenetrable, a determined intruder may try to cut a hole in the wall with a chainsaw.
- In what order will the intruder try these attacks? Usually, from the easiest to the hardest. The weakest link will attract the greatest number of attacks.



- For example, securing a credit card number should also include securing the system on which it resides, the network attached to that system, the other systems on the network, non-computer equipment (such as fax machines and phone switches) attached to that network, and the physical devices for each of these.
- Securing the data means discovering its path throughout the system and protecting it at every point.
- If the credit card number is stored on the most secure network but a business process that prints the card numbers and stores them is kept in an unlocked room, the attacker will exploit this weakest link.
- In a computer network, firewalls are often the strongest point of defense. They encounter their fair share of attacks, but most attackers know that properly configured firewalls are difficult to penetrate, so they will look for easier prey.
- This can take the form of lines in labs or small offices that aren't firewalled, modems and other remote access systems, Private Branch Exchange (PBX) phone switches, home computers and laptops that are sometimes connected to the company network, unpatched web servers and other Internet-facing servers, e-mail servers, and Domain Name Service (DNS) servers that are accessible from the Internet.
- All these typically offer less resistance to attackers than firewalls offer. That's why the security of these objects needs to be equally as strong as the firewall.

1.8 JUSTIFYING SECURITY INVESTMENT

- Specific benefits of a strong security program are business agility, cost reduction, and portability.

a. Business Agility:

- Every company wants to open up its business operations to its customers, suppliers, and business partners, in order to reach more people and facilitate the expansion of revenue opportunities.
- For example, manufacturers want to reach individual customers and increase sales through e-commerce websites.
- Weak security leaves many companies blind to the daily flow of information to and from their infrastructure. Security allows information to be used more effectively in advancing the goals of the organization because that organization can safely allow more outside groups of people to utilize the information when it is secure.
- The more access you provide; the more people you can reach.

b. Cost Reduction:

- Modern security practices do reduce some costs, such as those resulting from the loss of data or equipment. Data loss due to mishandling, misuse, or mistakes can be expensive.
- An extensive virus outbreak, a website, or a denial of service (DoS) attack can result in service outages during which customers cannot make purchases and the company cannot transact business.
- An increasing number of attacks are categorized as advanced persistent threats (APTs). These attacks are designed to deploy the malware into a network and remain undetected until triggered for some malicious purpose.
- Often, the goal of the attacks is the theft of financial information or intellectual property. Loss of service or leakage of sensitive data can result in fines, increased fees, and an overall decrease in corporate reputation and stock price.
- Strong security reduces loss of information and increases service availability and confidentiality.

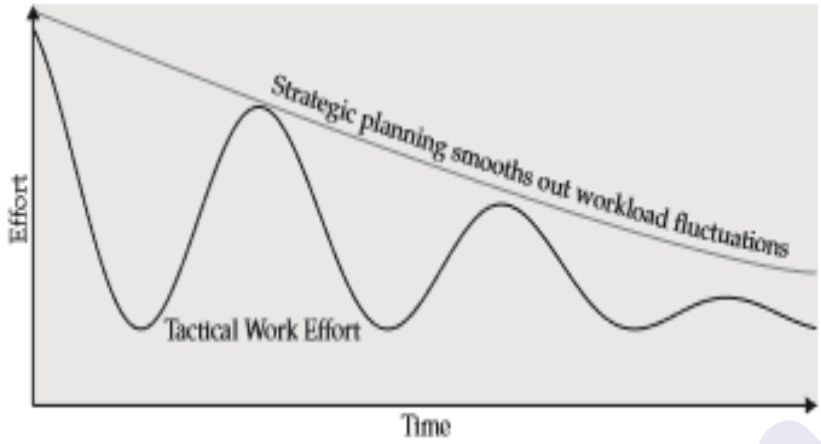
c. Portability:

- Portability means that software and data can be used on multiple platforms or can be transferred/transmitted within an organization, to a customer, or a business partner.
- The "consumerization" of information has placed demands on companies to be able to provide meaningful and accurate information at a moment's notice.
- Portability also enables business and creates value.
- For example, Apple's ability to both host music and allow personal music libraries to be synchronized to a tablet, mobile phone, and MP3 player.

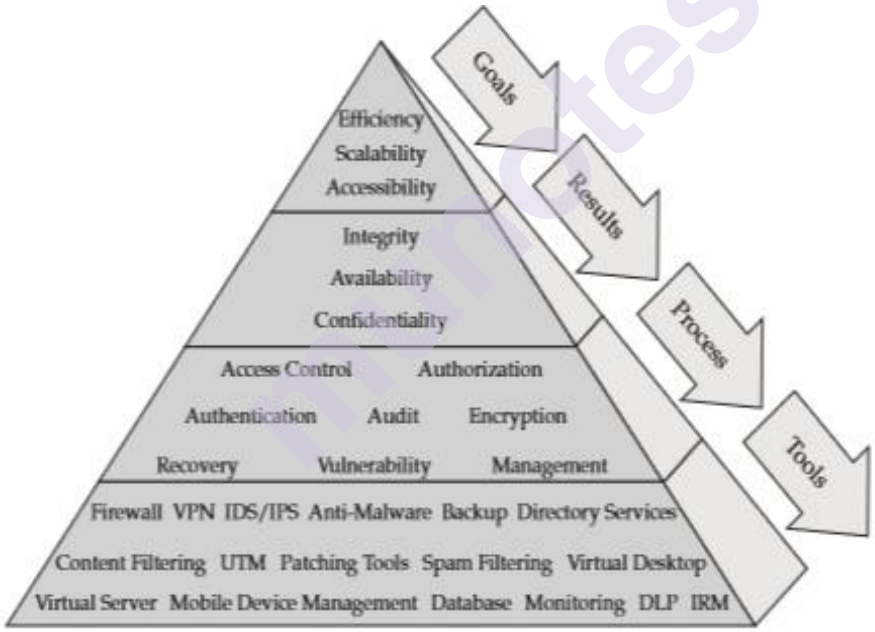
1.9 STRATEGY AND TACTICS

- A security strategy is the definition of all the architecture and policy components that make up a complete plan for defense, detection, and deterrence.
- Security tactics are the day-to-day practices of the individuals and technologies assigned to the protection of assets.
- Strategies are usually proactive, and tactics are often reactive. Both are equally important, and a successful security program needs to be both strategic and tactical in nature.
- Strategic planning can proceed on a weekly, monthly, quarterly, and yearly basis.

- If a company finds itself focusing only on strategy or only on tactics, it should review its priorities and consider adding additional staff to address the shortfall.
- Figure demonstrates the interplay of strategy and tactics.



1.10 BUSINESS PROCESSES VS. TECHNICAL CONTROLS



1.11 SUMMARY

- Information is an important asset.
- Information can be classified into different categories. This is typically done to control access to the information in different ways, depending on its importance, its sensitivity, and its vulnerability to theft or misuse.
- The field of security is concerned with protecting assets in general. Information security is concerned with protecting information in all

its forms, whether written, spoken, electronic, graphical, or using other methods of communication.

- Network security is concerned with protecting data, hardware, and software on a computer network.

1.12 QUESTIONS

1. Differentiate data and information.
2. What is security? Why it is needed.
3. Differentiate w.r.t security Government perimeter blockade model & Academic world.
4. What are the three D's of security?
5. What are the components needed to build a security program?
6. Justify the statement "An impossible job: The job of the attacker is always easier than the job of the defender".
7. Brief about the weakest link in security.
8. Explain the benefits of a strong security program: business agility, cost reduction, and portability.
9. What is meant by Strategy and Tactics?

1.13 REFERENCES

- "The complete reference: Information Security, Second Edition, By Mark Rhodes-Ousley".
- All contents are taken from this book.

*****

2

RISK ANALYSIS, SECURE DESIGN PRINCIPLES

Unit Structure

2.0 OBJECTIVES

The objectives are:

- To study various security breaches & attacks happening in an organization day by day.
- To prevent, and protect against such attacks with advanced tools & techniques.

2.1 DEFINITION

- Security professionals know that many real-world threats come from inside the organization, which is why building a wall around your

trusted interior is not good enough. You need to make sure your security controls focus on the right threats.

2.2 INTRODUCTION

- Any computer that is accessible from the Internet will be attacked. It will constantly be probed by attackers and malicious programs intending to exploit vulnerabilities.

2.3 RISK ANALYSIS

- The objective of a security program is to mitigate risks.
- Mitigating risks does not mean eliminating them, it means reducing them to an acceptable one to make sure your security controls are effectively controlling the risks in your environment.
- One needs to anticipate what kinds of incidents may occur and also needs to identify what you are trying to protect, and from whom.

2.4 THREAT DEFINITION

There are important threats and attacks given as follows:

- Threat vectors
- Threat sources and targets
- Types of attacks
- Malicious mobile code
- Advanced Persistent Threats (APTs)
- Manual attacks

2.5 THREAT VECTORS

- A threat vector is a term used to describe where a threat originates and the path it takes to reach a target. An example of a threat vector is an e-mail message sent from outside the organization to an inside employee, containing an irresistible subject line along with an executable attachment that happens to be a Trojan program, which will compromise the recipient's computer if opened.

Trojan programs are installed pieces of software that perform functions with the privileges of authorized users but are unknown to those users.

- Common functions of Trojans include stealing data and passwords, providing remote access and/or monitoring to someone outside the trusted network, or performing specific functions such as spamming.

- Trojans can be exploited over the Internet, through the firewall, or across the internal network by users who are not authorized to have access. Trojans are dangerous because they can hide in authorized communication channels such as web browsing.
- **Viruses** typically arrive in documents, executable files, and e-mail. They may include Trojan components that allow direct outside access, or they may automatically send private information, such as IP addresses, personal information, and system configurations, to a receiver on the Internet. These viruses usually capture and send password keystrokes as well.
- A further example is the **girlfriend exploit**. It refers to a Trojan program planted by an unsuspecting employee who runs a program provided by a trusted friend from a storage device like a disk or USB stick that plants a back door (also known as a trap door) inside the network.
- Another example is a malicious e-mail attachment that exploits the access rights of the person who opens the attachment to send confidential information out to the Internet.

2.6 THREAT SOURCES AND TARGETS

Security controls can be logically grouped into several categories:

- **Preventative:** Block security threats before they can exploit a vulnerability
- **Detective:** Discover and provide notification of attacks or misuse when they happen
- **Deterrent:** Discourage outsider attacks and insider policy violations
- **Corrective** Restore the integrity of data or another asset
- **Recovery** Restore the availability of a service
- **Compensative** In a layered security strategy, provide protection even when another control fails.

Each category of security control may have a variety of implementations to protect against different threat vectors:

- **Physical:** Controls that are physically present in the "real world"
- **Administrative:** Controls defined and enforced by management
- **Logical/technical:** Technology controls performed by machines
- **Operational:** Controls that are performed in person by people

- **Virtual:** Controls that are triggered dynamically when certain circumstances arise

2.7 TYPES OF ATTACKS

- When plain ASCII text was used to attack MS-DOS systems. It was possible because of a default-loaded device driver called **ansi.sys**, to create a plain-looking text file that was capable of remapping the keyboard.
- These malicious programs were called ANSI bombs. It was possible that after reading a text message, the next key pressed would format the hard drive—it did happen.
- Attacks can take the form of automated, malicious, mobile code traveling along networks looking for exploit opportunities or they can take the form of manual attempts by an attacker.
- An attacker may even use an automated program to find vulnerable hosts and then manually attack the victims, exploiting a single system vulnerability, which can compromise millions of computers in less than a minute.

Malicious Mobile Code:

There are three generally recognized variants of malicious mobile code: viruses, worms, and Trojans. In addition, many malware programs have components that act like two or more of these types, which are called hybrid threats or mixed threats.

The lifecycle of malicious mobile code looks like this:

1. Find
2. Exploit
3. Infect
4. Repeat

It just goes on every second of every day churning out replication cycles. Automated attacks are often very good at their exploit and only die down over time as patches close holes and technology passes them by.

2.8 COMPUTER VIRUSES

- A virus is a self-replicating program that uses other host files or code to replicate. Most viruses infect files so that every time the host file is executed, the virus is executed too.
- A virus infection is simply another way of saying the virus made a copy of itself (replicated) and placed its code in the host in such a way that it will always be executed when the host is executed.

- Viruses can infect program files, boot sectors, hard drive partition tables, data files, memory, macro routines, and scripting files.

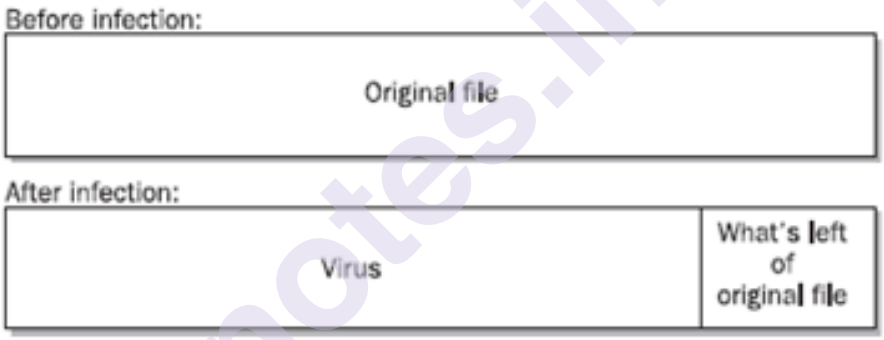
Anatomy of a Virus:

- The damage routine of a virus (or really of any malware program) is called the payload. The vast majority of malicious program files do not carry a destructive payload beyond the requisite replication.
- Payloads can be intentionally destructive, deleting files, corrupting data, copying confidential information, formatting hard drives, and removing security settings.
- Some viruses are devious. Many send out random files from the user's hard drive to everyone in the user's e-mail address list.
- There are even viruses that infect spreadsheets, changing numeric zeros into letter O's, making the cell's numeric contents become text and consequently, have a value of zero. The spreadsheet owner may think the spreadsheet is adding up the figures correctly, but the hidden O will make column and row sums add up incorrectly. This slowly corrupts all files on the hard drive.
- Viruses have been known to encrypt hard drive contents in such a way that if you remove the virus, the files become unrecoverable. A virus called **Caligula** even managed to prove that.
- A virus could steal private encryption keys. Viruses cannot break hard drive read-write heads, electrocute people, or cause fires. It happens when a virus focuses a single pixel on a computer screen for a very long time and causes the monitor to catch fire.

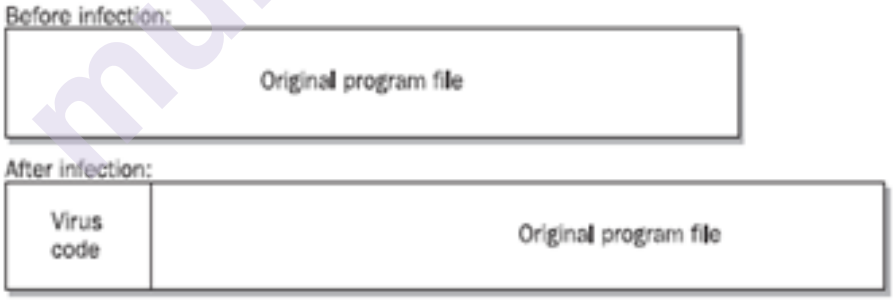
Types of Viruses:

- If the virus executes, does its damage, and terminates until the next time it is executed, it is known as a **non-resident virus**.
- A non-resident virus may, for example, look for and infect five EXE files on the hard disk and then terminate until the next time an infected file is executed. These types of viruses are easier for malicious coders to write.
- If the virus stays in memory after it is executed, it is called a **memory-resident virus. Memory**-resident viruses insert themselves as part of the operating system or application and can manipulate any file that is executed, copied, moved, or listed. Memory-resident viruses are also able to manipulate the operating system to hide from administrators and inspection tools. These are called **stealth viruses**.
- Other stealth viruses will hide the increase in file size and memory incurred because of the infection, make the infected file invisible to disk tools and virus scanners, and hide file modification attributes.

- If the virus overwrites the host code with its own code, effectively destroying much of the original contents, it is called an **overwriting virus.**
- If the virus inserts itself into the host code, moving the original code around so the host programming remains and is executed after the virus code, the virus is called a **parasitic virus**.
- Viruses that copy themselves to the beginning of the file are called **prepending viruses**, and viruses placing themselves at the end of a file are called **appending viruses**. Viruses appearing in the middle of a host file are labelled **mid-infecting viruses**.
- The modified host code doesn't always have to be a file—it can be a disk boot sector or partition table, in which case the virus is called a **boot sector or partition table virus**, respectively.
- For a pure boot sector virus to infect a computer, the computer must have booted, or attempted to boot, an infected disk.



Example of an overwriting virus



Example of a prepending parasitic virus

- Some boot sector viruses, like Tequila, are classified as **multipartite viruses** because they can infect both boot sectors and program files.
- If activated in their executable file form, they will attempt to infect the hard drive and place the infected boot code without having been transferred from an infected booted disk.
- Boot sector viruses move the original operating system boot sector to a new location on the disk, and partition table viruses manipulate the disk partition table in order to gain control first.

- Most boot sector virus damage routines run at the beginning of the virus's execution before Windows is loaded. The virus can damage Windows by preventing it from loading or by formatting the hard drive.
- **Macro viruses** infect the data running on top of an application by using the program's macro or scripting language.

2.9 COMPUTER WORMS

- A computer worm uses its own coding to replicate, although it may rely on the existence of other related code. The key to a worm is that it does not directly modify other host codes to replicate.
- A worm may travel the Internet trying one or more exploits to compromise a computer, and if successful, it then writes itself to the computer and begins replicating again.
- An example of an Internet worm is Bugbear. Bugbear was released in June 2003, arriving as a file attachment in a bogus e-mail.
- It adds itself to the Windows start-up group so it gets executed each time Windows starts. Bugbear looks for and attempts to gain access to weakly password-protected network shares and terminates antivirus programs.

E-Mail Worms:

- E-mail worms are the intersection of social engineering. They appear in people's inboxes as messages and file attachments from friends, strangers, and companies. They pose as cute games, official patches from Microsoft, or unofficial applications found in the digital marketplace.
- The worm first modifies the PC in such a way that it makes sure it is always loaded into memory when the machine starts.
- Then it looks for additional e-mail addresses to send itself to. It might use Microsoft's Messaging Application Programming Interface (MAPI) or use the registry to find the physical location of the address book file.

2.10 TROJANS

- Trojan horse programs, or Trojans, work by posing as legitimate programs that are activated by an unsuspecting user. After execution, the Trojan may attempt to continue to pose as the other legitimate program (such as a screensaver) while doing its malicious actions in the background.
- Many people are infected by Trojans for months and years without realizing it. If the Trojan simply starts its malicious actions and doesn't pretend to be a legitimate program, it's called a **direct-action**

Trojan. Direct-action Trojans don't spread well because the victims notice the compromise and are unlikely, or unable, to spread the program to other unsuspecting users.

Remote Access Trojans:

- A powerful type of Trojan program called a Remote Access Trojan (RAT) is very popular.
- Once installed, a RAT becomes a back door and allows the remote attackers to do virtually anything they want to the compromised PC.
- RATs can delete and damage files, download data, manipulate the PC's input and output devices, and record keystroke screenshots and screen-capturing.
- It allows the attacker to track what the user is doing, including the entry of passwords and other sensitive information. If the compromised user visits their bank's website, the attacker can record their login information.
- RATs have even been known to record video and audio from the host computer's web camera and microphone. Imagine malware that is capable of recording every conversation made near the PC.

Zombie Trojans and DDoS Attacks:

- Zombie Trojans infect a host and wait for their originating attacker's commands telling them to attack other hosts. The attacker installs a series of zombie Trojans.
- With one predefined command, the attacker can cause all the zombies to begin to attack another remote system with a distributed denial of service (DDoS) attack.
- DDoS attacks flood the intended victim's computer with so much traffic, legitimate or malformed, that it becomes overutilized or locks up, denying legitimate connections. Zombie Trojan attacks have been responsible for some of the most publicized attacks on the Internet, temporarily paralyzing targets like Yahoo, eBay, Microsoft, Amazon, and the Internet's DNS root servers.

2.11 MALICIOUS HTML

The Internet allows for many different types of attacks, many of which are HTML-based Pure HTML coding can be malicious when it breaks browser security zones or when it can access local system files.

For example, the user may believe they are visiting a legitimate website, when in fact an attacker has hijacked their browser session and the user is inputting confidential information into an attacker's site. Malicious HTML has often been used to access files on local PCs, too. Specially crafted

HTML links can download files from the user's workstation, retrieve passwords, and delete data.

2.12 ADVANCED PERSISTENT THREATS (APTS)

- The use of sophisticated malware for targeted cybercrime is known as advanced persistent threats (APTs). APTs are created and directed by hostile governments and organized criminals for financial or political gain.
- APTs are intentionally stealthy and difficult to find and remove—they may hide for months on an organization's network doing nothing until they are called upon by their controllers.
- Once the malware infects the victim's computer, usually silently and without the user's knowledge, it "phones home" to download further malware.
- In this second phase of the attack, the malware reaches out to a command and control server (CnC server) to bring down rootkits,
- Trojans, RATs, and other sophisticated malware—in effect, completely compromise the victim's computer and usually without any indication that anything is wrong.
- APTs use the very latest infection techniques against newly discovered vulnerabilities. Finally, in the third phase of the attack, the RATs open up connections to their CnC servers, to be used by their human controllers at their leisure. When malicious operators take over the victim's computer, they have full access to everything inside the organization that the user has access to.
- This can be a targeted attack against a victim within the organization, such as an engineer or researcher with access to confidential material. The attacker may send an infected document, such as a PDF file, to the victim, along with a highly believable e-mail message to trick the victim into opening the file.
- Alternatively, the attacker may send a URL that points to a web server that executes malicious Java or ActiveX code on the victim's browser—even without the victim's intervention. This is known as a drive-by download.

Manual Attacks:

While automated attacks may satisfy virus writers, typical attackers want to test their own mental wits and toolkits against a foreign computer, changing their attack plan as the host exposes its weaknesses.

Physical Attacks:

Another means of attack is direct physical access, but if an attacker can physically access a computer, it's game over. They literally can do

anything, including physically damaging the computer, and stealing passwords and data.

In some cases, the attacker may first compromise a legitimate website the victim may run across during normal business research, or poison DNS entries to send the victim to their compromised website.

In either case, the malicious code is run by the victim's web browser without requiring the user to respond "Yes" or "Continue" to any prompts. All of these targeted attacks are collectively known as spear-phishing.

2.13 NETWORK-LAYER ATTACKS

Many attacker attacks are directed at the lower six layers of the Open Systems Interconnection (OSI) network protocol model. Network-layer attacks include packet sniffing and protocol-anomaly exploits.

Packet Sniffing:

Sniffing occurs when an unauthorized third party captures network packets destined for computers other than their own. Packet sniffing allows the attacker to look at transmitted content and may reveal passwords and confidential data.

Specialized packet driver software, must be connected to the network segment they want to sniff, and must use sniffer software. By default, a network interface card (NIC) in a computer will usually drop any traffic not destined for it. By putting the NIC in promiscuous mode, it will read any packet going by it on the network wire.

Packet-sniffing attacks are more common in areas where many computer hosts share the same collision domain.

Protocol-Anomaly Attacks:

Network-layer attacks usually require that the attacker create malformed traffic, which can be created by tools called packet injectors or traffic generators. Packet injectors are used by legitimate sources to test the throughput of network devices or to test the security defences of firewalls and IDSs.

Attackers can even manually create the malformed traffic as a text file and then send it using a traffic replay tool.

2.14 APPLICATION-LAYER ATTACKS

Application-layer attacks include any exploit directed at the applications running on top of the OSI protocol stack. Application-layer attacks include exploits directed at application programs, as well as against operating systems. Application-layer attacks include content attacks, buffer overflows, and password-cracking attempts.

Buffer Overflows:

Buffer overflows occur when a program expecting input does not do input validation

Password Cracking:

Password crackers either try to guess passwords or use brute-force tools. Brute-force tools attempt to guess a password by trying all the character combinations listed in an accompanying dictionary. The dictionary may start of blindly guessing passwords using a simple incremental algorithm. (example, trying aaaaa, aaaab, aaaac, and so on) or it may use passwords known to be common on the host (such as password, blank, michael, and so on).

If the attacked system locks out accounts after a certain number of invalid login attempts, some password attackers will gain enough access to copy down the password database, and then brute-force it offline.

P2P Attacks:

With the advent of peer-to-peer (P2P) services, malicious programs are spreading from PC to PC without having to jump on e-mail or randomly scan the Internet for vulnerabilities. No matter how the attack occurs, whether automated or manual, most exploits are only successful on systems without basic countermeasures installed.

Man-in-the-Middle Attacks:

- Man-in-the-middle (MITM) attacks are a valid and extremely successful threat vector.
- A MITM attack can take a few different forms. ARP poisoning is the most common, but DHCP, DNS, and ICMP poisoning are also effective, as well as the use of a malicious wireless access point (AP).
- Fake APs have become a common threat vector, exploiting the manner in which clients automatically connect to known SSIDs. This enables an attacker to connect and intercept the victim's network traffic without the victim seeing any indication they are under attack. To hasten a connection, attacks against the legitimate AP can be made to help the malicious AP become the last AP standing.

ARP Poisoning:

- ARP poisoning works by simply responding to Address Resolution Protocol (ARP) requests with the attacker's MAC address. The attacker tells the device that wishes to communicate with the victim's computer that the attacker knows how to reach the victim, and then the attacker tells the network that the attacker's computer is the victim's computer—effectively masquerading (pretending) as the victim's computer and responding on its behalf.

- The switch then updates its table of MAC addresses with the attacker's MAC address. The switch uses this to route traffic and now believes the attacker's system is the victim's system.
- An ARP poisoning attack can be executed so that it only updates the ARP table of the victim and not the gateway (one-way poison). Many organizations protect the network Architecture.

MAC Flooding:

Technically known as MAC address flooding, is where an application injects a specially crafted layer two and layer three packet onto the network repeatedly. This causes the layer to switch to fill up its buffers and crash. Since the switch crash behavior is to fail/open, all ports are flooded with all frames, thus causing the denial of service.

DHCP Poisoning:

This attack allows an attacker to compromise victims with three simple steps: provide the pool of addresses to assign for the victims, provide the netmask for the victims, and finally provide the DNS IP address.

DNS Spoofing Attack:

A DNS spoofing attack is just as easy to execute as a DHCP poisoning attack. All traffic from the victim is forwarded through the attacker's fake DNS service and redirected so that all requests for Internet or internal sites land at the attacker's site, from which the attacker can harvest credentials or possibly launch browser-based attacks, such as Java runtime error, to trick the victim.

This can also be done through the local "hosts" file on the computer. The fundamentals of this attack come from "name resolution order" and manipulating that process. DNS is designed so that every DNS query first goes to a DNS server, usually a local one on the network or provided by the ISP.

That server will have been pre-configured with the IP addresses of the top-level (root) DNS servers on the Internet that are the authoritative "source of truth" for all IP addresses and hostnames. The root server that responds would respond with the address of a lower-level DNS server. This process continues until the name and IP address are found, usually at least three levels down.

ICMP Poisoning:

- The attacker wishing to execute an ICMP attack is that they need to be able to see all traffic; if they are attached to a switch, this attack is not useful because this is a layer three attack unless the attacker's computer is connected to a spanning port, which in turn would forward all traffic to the attacker's system so they could see it.

- Simple, easy-to-use attack tools are available on the Internet that automates the attack. An attacker only has to provide the MAC address of the gateway and the IP address of the gateway. The attack tool will do the rest.

2.15 RISK ANALYSIS

Risk analysis needs to be a part of every security effort. It should analyze and categorize the assets that need to be protected and the risks that need to be avoided, and it should facilitate the identification and prioritization of protective elements.

It can also provide a means to measure the effectiveness of the overall security architecture, by tracking those risks and their associated mitigation over time to observe trends.

- **Risk = Probability (Threat + Exploit of Vulnerability) * Cost of Asset Damage.**
- One commonly used approach to assigning a cost to risks is annualized loss expectancy (ALE).
- This is the cost of an undesired event—a single loss expectancy (SLE)—multiplied by the number of times you expect that event to occur in one year—the annualized rate of occurrence (ARO).
- **Annualized Loss (ALE) = Single Loss (SLE) * Annualized Rate (ARO)**

2.16 THE CIA TRIAD AND OTHER MODELS

Confidentiality:

- Confidentiality refers to the restriction of access to data only to those who are authorized to use it. This means a single set of data is accessible to one or more authorized people or systems, and nobody else can see it.
- Confidentiality is distinguishable from privacy in the sense that "confidential" implies access to one set of data by many sources, while "private" usually means the data is accessible only to a single source.
- As an example, a password is considered private because only one person should know it, while a patient record is considered confidential because multiple members of the patient's medical staff are allowed to see it.

Integrity:

- Integrity, which is particularly relevant to data, refers to the assurance that the data has not been altered in an unauthorized way.

- Integrity controls are meant to ensure that a set of data can't be modified (or deleted entirely) by an unauthorized party.
- Part of the goal of integrity controls is to block the ability of unauthorized people to make changes to data, and another part is to provide a means of restoring data back to a known good state.

Availability:

- Availability refers to the "uptime" of computer-based services—the assurance that the service will be available when it's needed. Service availability is usually protected by implementing high-availability (or continuous-service) controls on computers, networks, and storage. High-availability (HA) pairs or clusters of computers, redundant network links, and RAID disks are examples of mechanisms to protect availability.
- The best-known attributes of security defined in the preceding models and others like them include Confidentiality, Integrity, Availability, Accountability, Accuracy, Authenticity, Awareness, Completeness, Consistency, Control Democracy, Ethics, Legality, Non-repudiation, Ownership, Physical Possession, Reassessment, Relevance, Response, Responsibility, Risk Assessment Security Design and Implementation, Security Management, Timeliness, Utility.

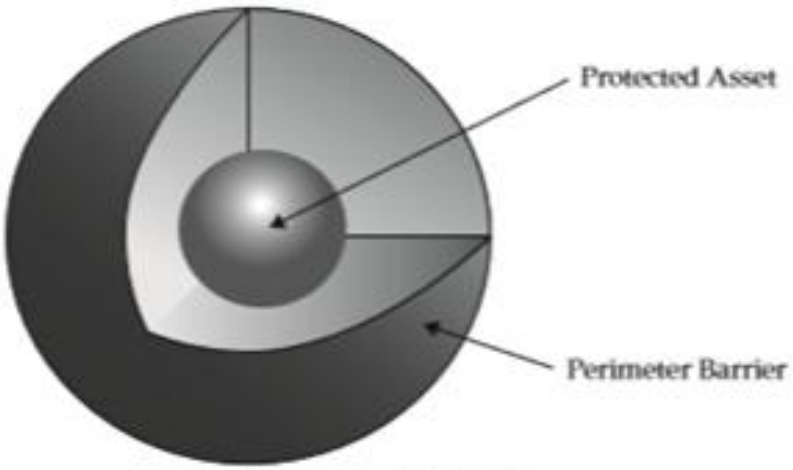
2.17 DEFENSE MODELS

- There are two approaches to preserving the confidentiality, integrity, availability, and authenticity of electronic and physical assets such as the data on your network:
- Build a defensive perimeter around those assets and trust everyone who has access inside.
- Use many different types and levels of security controls in a layered defense-in-depth approach.

a. The Lollipop Model:

- The most common form of defense, known as perimeter security, involves building a virtual (or physical) wall around objects of value. Perimeter security is like a lollipop with a hard, crunchy shell on the outside and a soft, chewy center on the inside.
- Consider the example of a house—it has walls, doors, and windows to protect what's inside (a perimeter). But does that make it impenetrable? No, because a determined attacker can find a way in—either by breaking through the perimeter exploiting some weakness in it, or convincing someone inside to let them in.
- By comparison, in network security, a firewall is like a house—it is a perimeter that can't keep out all attackers.

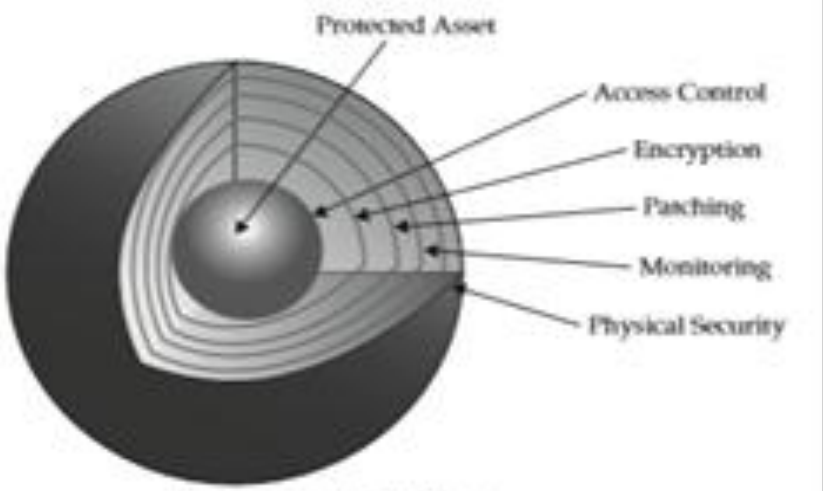
- The firewall is the most common choice for controlling outside access to the internal network, creating a virtual perimeter around the internal network.



The lollipop model of defense

b. The Onion Model:

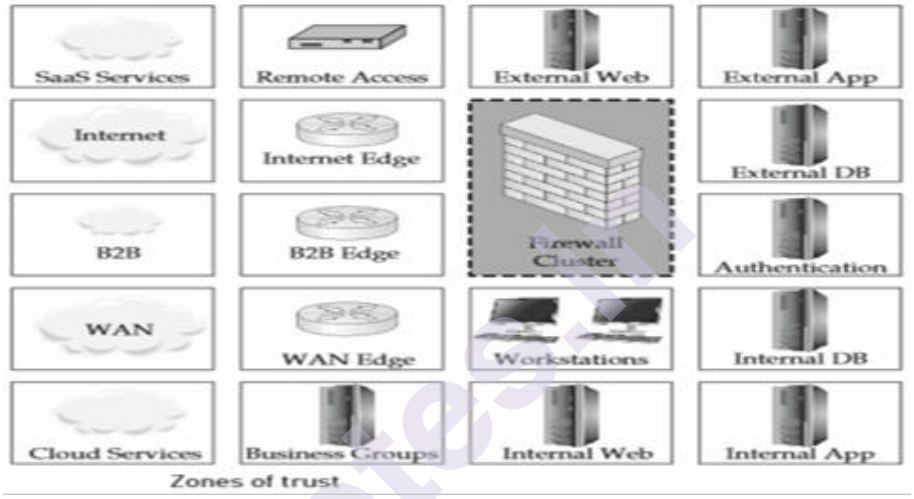
- It is a layered strategy, often referred to as defense in depth. This model addresses the contingency of a perimeter security breach occurring. It includes the strong wall of the lollipop.
- A layered security architecture, like an onion, must be peeled away by the attacker, layer by layer, with plenty of crying.
- The more layers of controls that exist, the better the protection against a failure of any one of those layers.
- The layered security approach can be applied at any level where security controls are placed, not only to increase the amount of work required for an attacker to break down the defenses but also to reduce the risk of unintended failure of any single technology.



The onion model of defense

2.18 ZONES OF TRUST

Different areas of a network trust each other in different ways. Some communications are trusted completely—the services rely on assumptions that the sender and recipient are on the same level as if they were running on a single system. Some are trusted incompletely— they involve less trusted networks and systems, so communications should be filtered. Some networks (like the Internet or wireless hot spots) are untrusted. The security controls should carefully screen the interfaces between each of these networks. These definitions of trust levels of networks and computer systems are known as zones of trust.



- Zones of trust are connected with one another, and business requirements evolve and require communications between various disparate networks, systems, and other entities on the networks.
- The use of multiple zones allows access between a less and a more trusted zone to be controlled to protect a more trusted resource from attack by a less trusted one.
- The importance of trust models is that they allow a broad, enterprise-wide view of networks, systems, and data communications, and they highlight the interactions among all of these components.
- Trust can also be viewed from a transaction perspective. During a particular transaction, several systems may communicate through various zones of trust. In a transaction-level trust model, instead of systems being separated into different trust zones based on their locations on the network (as is done with the Internet, a DMZ, and an internal network), systems can be separated into functional categories based on the types of transactions they process.
- For example, a credit card transaction may pass through a web server, an application server, a database, and a credit-checking service on the Internet. During the transaction, all of these systems must trust each other equally, even though the transaction may cross several network boundaries. Thus, security controls at the system and network levels

should allow each of these systems to perform their authorized functions while preventing other systems not involved in the transaction from accessing these resources.

- Segmenting network data resources based on their access requirements is a good security practice. Segmentation allows greater refinement of access control based on the audience for each particular system, and it helps confine the communications between systems to the services that have transactional trust relationships.

- A layered segmentation approach also provides a useful conceptual model for network and system administrators. Several groups of servers can be included in a layer, defined by the types of services they perform, the types of data they handle, and the places they need to communicate to and from.

- For example, a public layer may contain systems that accept communication directly from the Internet. An application layer may contain systems that accept communication from the public layer.

- A data layer may accept communication from the application layer. Communication between these layers can be managed by a firewall, or by ACLs.

Best Practices for Network Defense:

- There are many countermeasures you can implement to minimize the risk of a successful attack, such as securing the physical environment, hardening the operating systems, keeping patches updated, using an antivirus scanner, using a firewall, securing network share permissions, using encryptions, securing applications, backing up the system, creating a computer security defense plan, and implementing ARP poisoning defenses.

Secure the Physical Environment:

- Regular PCs need physical protection. Depending on the environment, PCs and laptops might need to be physically secured to their desks.

- There are several different kinds of lockdown devices. If anyone leaves their laptop on their desk overnight, it should be secured. There are also other steps that need to be taken on every PC in your environment.

Password Protect Booting:

- Consider requiring a boot-up password before the operating system will load. This can usually be set in the CMOS/BIOS and is called a user or boot password. This is especially important for portable computers, such as laptops and tablets, and smartphones.

Password Protect CMOS:

- The CMOS/BIOS settings of a computer contain many potential security settings, such as boot order, remote wake-up, and antivirus boot-sector protection. It is important to ensure that unauthorized users do not have access to the CMOS/BIOS settings.
- Most CMOS/BIOSs allow you to set up a password to prevent unauthorized changes. The password should not be the same as other administrative passwords, but for simplicity's sake, a common password can be used for all machines.

Harden the Operating System:

- Reduce the attack surface of the operating system by removing unnecessary software, disabling unneeded services, and locking down access.
- Reduce the attack surface of systems by turning off unneeded services.
- Install secure software.
- Configure software settings securely.
- Patch systems regularly and quickly.
- Segment the network into zones of trust and place systems into those zones based on their communication needs and Internet exposure.
- Strengthen authentication processes.
- Limit the number (and privileges) of administrators

Keep Patches Updated:

- A solid patch management plan is essential for protecting any platform, regardless of operating system and regardless of whether or not it is connected directly to the Internet

Use an Antivirus Scanner (with Real-Time Scanning):

- In today's world, an antivirus (AV) scanner is essential. It should be deployed on your desktop, with forced, automatic updates, and it should be enabled for real-time protection.
- By placing the antivirus solution on the desktop, you are ensuring that no matter how it gets there, it will be blocked. E-mail and gateway AV are the only solutions that work most of the time, but they will fail if the malware comes in via any other method or on an unexpected port.
- The AV solution should be enabled for real-time protection so it scans every file as it comes into the system or enters the computer's memory, so it can prevent malware from executing.

Use Firewall Software:

- Firewalls have come a long way since their days of simple port filtering. Today's devices are stateful inspection systems capable of analyzing threats occurring anywhere in layers three through seven with software that runs directly on the computer.
- Firewalls are able to collate separate events into one threat description and can identify the attack by name. Every PC should be protected by firewall software. Desktop firewall software can protect a PC against internal and external threats and usually offer the added advantage of blocking unauthorized software applications (such as Trojans) from initiating outbound traffic.

Secure Network Share Permissions:

- Folders and files accessed remotely over the network should have discretionary ACLs (DACLs) applied using the principle of least privilege and should have complex passwords. By default, Windows assigns, and most administrators allow, the Everyone group to have Full Control or Read permissions throughout the operating system and on every newly created share.
- A better strategy is to assign share and NTFS permissions to the smallest allowable list of groups and users. That way, if you accidentally set your NTFS file permissions to open, the share permissions might counteract the mistake.

Use Encryption:

- Encrypting File System (EFS) is one of the most exciting features in Windows. EFS encrypts and decrypts protected files and folders on the fly. Once turned on by a user, EFS will automatically generate public/private encryption key pairs for the user and the recovery agent. All the encrypting and decrypting are done invisibly in the background. If an unauthorized user tries to access an EFS-protected file, they will be denied access.
- It won't prevent malware occurrences while the authorized user is logged on. However, EFS-protected folders and files will be protected when the authorized user is not logged on. EFS can help provide additional security and is virtually invisible to the end user.

Secure Applications:

- Managing your applications and their security should be a top priority for any administrator. Applications can be managed by configuring application security, installing applications to nonstandard directories and ports, locking down applications, securing P2P services, and making sure your application programmers code securely.

Securely Configure Applications:

- Applications should be configured with the vendors' recommended security settings. In end-user PC environments, however, you want to keep the applications and minimize the risk at the same time. You can do this by regularly applying security patches and making sure security settings are set at the vendor's recommended settings, if not higher.

- Outlook and Outlook Express should both have their security zone set to Restricted. Internet Explorer's Internet zone should be set to Medium-High or High. The office offers administrative templates (called ADM files) that can be configured and deployed using System Policies or Group Policies.

Securing E-Mail

- E-mail worms continue to be the number one threat to computer systems. Most worms arrive as a file attachment or as an embedded script that the end user executes. You can significantly decrease your network's exposure risk by securing e-mail. This can be done by disabling HTML content and blocking potentially malicious file attachments.

- For that reason, it is important to restrict e-mails to plain text only or, if you must allow it, plain HTML coding only. You should disable scripting languages and active content, such as ActiveX controls, Java, and VBScript objects.

Secure P2P Services

- Peer-to-peer (P2P) applications, like instant messaging (IM) and music sharing, are likely to remain strong attack targets in the future. This is because P2P applications have very limited security, if any, and are often installed in the corporate environment without the administrator's authorization. And, they are designed to access files on the end user's computer, which makes the job of stealing those files that much easier. A firewall is configured to explicitly stop P2P traffic.

a. Implement Static ARP Tables:

- From a console, if you execute the command arp –a, it will display the ARP table for your system. This is how the system knows how to route traffic.

- This is the address for the switch where traffic will pass if the device wants to send information to a device that doesn't exist in its ARP table. A simple ARP request is sent to ask for the information.

- The information is then added to the ARP table of the device. The switch follows the same steps to build its ARP table. This is known as dynamic updating

- Static ARP tables, instead of using the basic ARP request/reply method, the tables are managed by the organization, and essentially hard coded. This helps to prevent an ARP poisoning attack.

b. Configure Port Rate Limiting:

- In this scenario, the amount of traffic passing over a port during a given length of time is monitored. If the configured threshold is tripped, the port closes itself until either it is enabled manually or a specified length of time passes (usually 15 minutes).
- In order to establish an effective threshold, an organization will need to monitor the amount of traffic for a "normal" system. By monitoring traffic correctly, a proper threshold can be set. If the organization does not do its research ahead of time and simply implements what it thinks is a "good" threshold, it may find that its users are constantly exceeding the threshold and unable to perform their day-to-day work.
- In order to establish an effective threshold, an organization will need to monitor the amount of traffic for a "normal" system over the course of a few weeks. By monitoring traffic correctly, a proper threshold can be set.

c. Use DHCP Snooping and Dynamic ARP Inspection:

- The most effective defense against ARP poisoning is to use DHCP snooping with Dynamic ARP inspection (DAI). The basis of this defense is that it drops all ARP reply requests not contained within its table.
- The organization needs to run DHCP snooping for two to three weeks in order to build a proper table of IP addresses and MAC addresses. After it has built that table, it can implement DAI.

2.19 SUMMARY

In this chapter, various topics covered are Network layer attacks, Application level attacks, DNS, DHCP, etc.

2.20 QUESTIONS

1. What is meant by risk analysis?
2. What is a Threat? Define threat vector.
3. What is virus & Trojans?
4. Name seven categories of different security controls.
5. Explain different types of threat attacks.
6. Explain the following :
   a. Malicious Mobile Code

b. Computer Viruses

c. Computer Worms

d. E-Mail Worms

e. Trojans

f. Remote Access Trojans

g. Zombie Trojans

h. DDoS Attacks

i. Malicious HTML

7. Write about the Anatomy of a Virus.
8. What are the different types of viruses?
9. Write short notes on Advanced Persistent Threats (APTs)
10. What is meant by packet sniffing?
11. Explain the CIA Triad.
12. Describe the two types of defense models.
13. What is meant by zones of trust?
14. What are the Best Practices for Network Defense?
15. Briefly describe risk analysis.
16. Explain the following:
    a. Protocol-Anomaly Attacks
    b. Application-Layer Attacks
    c. Buffer Overflows attacks
    d. P2P Attacks
    e. Man-in-the-Middle Attacks
    f. ARP Poisoning
    g. MAC Flooding
    h. DHCP Poisoning
    i. DNS Spoofing Attack
    j. ICMP Poisoning

2.21 REFERENCES

- The complete reference: Information Security, Second Edition, By Mark Rhodes-Ousley. All contents are taken from this book.

*****

UNIT - II

3

AUTHENTICATION AND AUTHORIZATION, ENCRYPTION

Unit Structure

3.0 OBJECTIVES

- To understand Authentication
- To understand Authorization
- To learn Encryption

3.1 INTRODUCTION

One of the most common ways to control access to computer systems is to identify who is at the keyboard (and prove that identity), and then decide what they can access. This twin controls authentication and authorization respectively, ensuring that only authorized users get access to the computing resources while blocking access to unauthorized users.

Authentication is the means of verifying a person (or process) to whom they claim to be, while authorization determines what they're allowed to do. This should be done in accordance with the principle of least privilege—giving each person only the amount of access they require to be effective in their job function, and no more.

3.2 AUTHENTICATION

Authentication is the process by which people prove they are what they say they are. It consists of two parts: a public statement of identity (usually a username) combined with a private response to a challenge (such as a password). The secret response to the authentication challenge can be based on one or more factors

- something you know (a secret word, number, or passphrase for example)
- something you have (such as a smartcard, ID tag, or code generator) or
- something you are (like a biometric factor like a fingerprint or retinal print).

A password is a means of identifying someone through something only they should know, and it is the most common form of challenge-response and is an example of single-factor authentication. Single-factor authentication is the simplest form of authentication method. With SFA, a person matches one credential to verify him or herself online. This is not considered a strong authentication method because a password can be intercepted or stolen in a variety of ways—for example, passwords are frequently written down or shared with others, they can also be captured from the system or the network, and they are often weak and easy to guess. Year after year, studies find that passwords such as "123456," "password" and other poor passwords remain extremely popular. Imagine if you could only identify your friends by being handed a previously agreed secret phrase on a piece of paper instead of by looking at them or hearing their voices. How reliable would that be? This type of identification is often portrayed in spy movies, where a secret agent uses a password to impersonate someone the victim is supposed to meet but has never seen. This trick works precisely because it is so fallible—the password is the only means of identifying the individual. Passwords are just not a good way of authenticating someone. Unfortunately, password-based authentication was the easiest type to implement in the early days of computing, and the model has persisted to this day.

Other single-factor authentication methods are better than passwords. Tokens and smart cards are better than passwords because they must be in the physical possession of the user. Biometrics, which use a sensor or scanner to identify unique features such as fingerprints, facial recognition, hand geometry, iris recognition, and retinal identification of individual body parts are better than passwords because they can't be shared—the

user must be present to log in. However, there are ways to defeat these methods. Tokens and cards can be lost or stolen, and biometrics can be spoofed. Yet, it's much more difficult to do that than of stealing or obtaining a password. Passwords are the worst possible method of proving identity, despite being the most popular method.

Multifactor authentication refers to using two or more methods of checking identity. Multi-factor authentication (or MFA) is a multi-layered protection framework that verifies the login or other transaction identities of users. A few examples of multi-factor authentication are codes created by mobile apps, answers to personal security questions, codes sent to an email address, fingerprints, etc.

These methods include (listed in increasing order of strength):

- Something you know (a password or PIN code)
- Something you have (such as a card or token)
- Something you are (a unique physical characteristic)

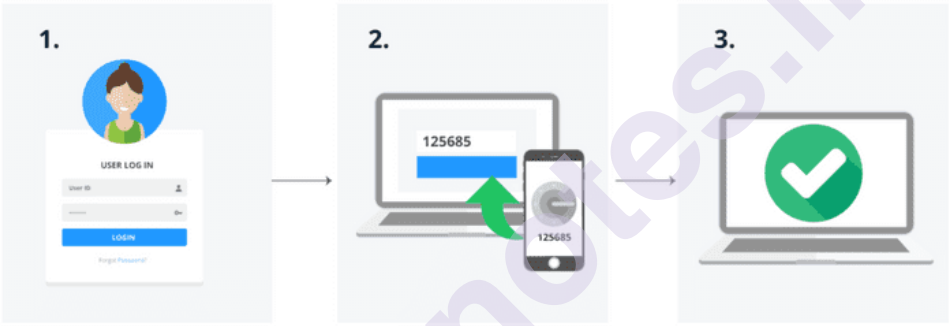


Figure 3.1 Multifactor Authentication

Two-factor authentication is the most common form of multifactor authentication, such as a password-generating token device with an LCD screen that displays a number (either time based or sequential) along with a password, or a smart card along with a password. Again, passwords aren't very good choices for a second factor, but they are ingrained into our technology and collective consciousness, they are built into all computer systems, and they are convenient and cheap to implement. A token or smart card along with biometrics would be much better—this combination is practically impossible to defeat. However, most organizations aren't equipped with biometric devices.

The following sections provide a detailed introduction to these types of authentication systems available today:

- Systems that use username and password combinations,
- Systems that use certificates or tokens
- Biometrics

3.2.1 Usernames and Passwords:

In the familiar method of password authentication, a challenge is issued by a computer and the user wishing to be identified provides a response. If the response can be validated, the user is said to be authenticated, and the user can access the system. Otherwise, the user is prevented from accessing the system.

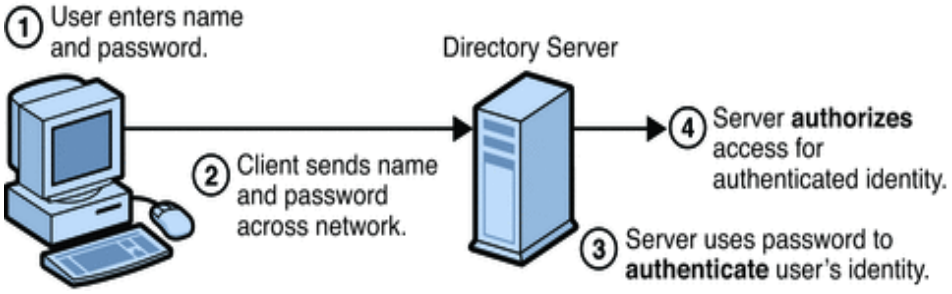


Figure 3.2 Username and password-based authentication

3.2.2 One-Time Password Systems:

Two problems plague passwords. First, they are (in most cases) created by people. Thus, people need to be taught how to create strong passwords, and most people aren't taught (or don't care enough to follow what they're taught). These strong passwords must also be remembered by a person and not written down. People do write passwords down and often leave them where others can find them. People commonly share passwords despite all your warnings and threats. Passwords are subject to a number of different attacks. They can be captured and cracked, or used in a replay attack in which the passwords are intercepted and later used to repeat authentication.

One solution to this type of attack is to use an algorithm that requires the password to be different every time it is used. In systems other than computers, this has been accomplished with the use of a one-time pad. When two people need to send encrypted messages, if they each have a copy of the one-time pad, each can use that password or some other method for determining which password to use. The advantage, of course, to such a system, is that even if a key is cracked or deduced, it is only good for the current message. The next message uses a different key.

One-Time Password as SMS Message:

Originally, most OTP's were sent as SMS messages. Once the user has begun his login attempt, filling in his/her username and the correct password, an SMS OTP is sent to the mobile number connected to his/her account. The user then enters the code shown on this phone in the login screen, completing the authentication process.

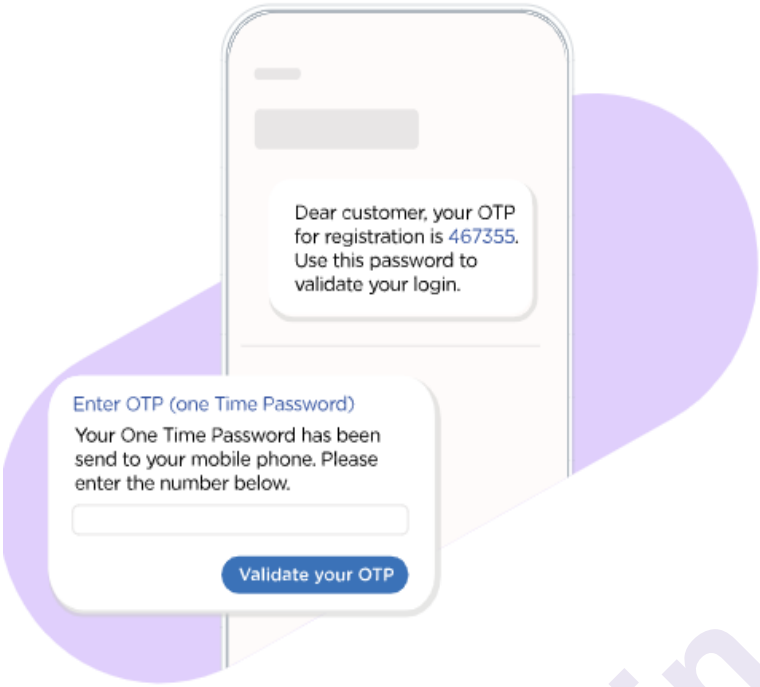


Figure 3.3 OTP Based authentication

3.2.3 Certificate-Based Authentication:

A certificate is a collection of information that binds an identity (user, computer, service, or device) to the public key of a public/private key pair. The typical certificate includes information about the identity and specifies the purposes for which the certificate may be used, a serial number, and a location where more information about the authority that issued the certificate may be found. The certificate is digitally signed by the issuing authority, the certificate authority (CA). The infrastructure used to support certificates in an organization is called the **Public Key Infrastructure (PKI)**.

The certificate, in addition to being stored by the identity it belongs to, may itself be broadly available. It may be exchanged in e-mail, distributed as part of some application's initialization, or stored in a central database of some sort where those who need a copy can retrieve one. Each certificate's public key has its associated private key, which is kept secret, and usually only stored locally by the identity. (Some implementations provide private key archiving, but often it is the security of the private key that provides the guarantee of identity.) Public/Private key algorithms use two keys: one key is used to encrypt, the other to decrypt. If the public key encrypts, only the related private key can decrypt. If the private key encrypts, only the related public key can decrypt.

When certificates are used for authentication, the private key is used to encrypt or digitally sign some request or challenge. The related public key (available from the certificate) can be used by the server or a central authentication server to decrypt the request. If the result matches what is expected, then proof of identity is obtained. These authentication steps are given as follows:

1. The client issues an authentication request.
2. A challenge is issued by the server.
3. The workstation uses its private key to encrypt the challenge.
4. The response is returned to the server.
5. Since the server has a copy of the certificate, it can use the public key to decrypt the response.
6. The received result is compared to the challenge.
7. If there is a match, the client is authenticated.

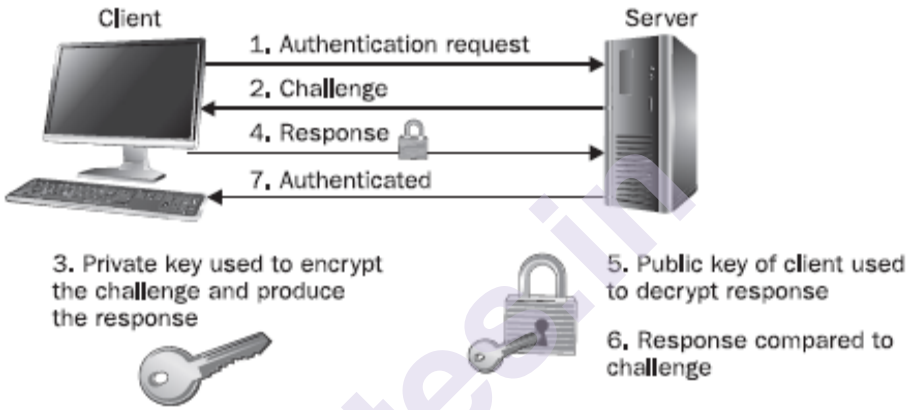


Figure 3.4 Certificate authentication uses public and private keys

It is useful here to understand that the original set of keys is generated by the client, and only the public key is sent to the CA. The CA generates the certificate and signs it using its private key, and then returns a copy of the certificate to the user and to its database. In some systems, another database also receives a copy of the certificate. It is the digital signing of the certificate that enables other systems to evaluate the certificate for its authenticity. If they can obtain a copy of the CA's certificate, they can verify the signature on the client certificate and thus be assured that the certificate is valid.

3.2.3.1 SSL/TLS:

Secure Sockets Layer (SSL) is a digital certificate system that is used to provide authentication of secure web servers & clients and to share encryption keys between servers and clients. SSL is a security protocol that creates an encrypted link between a web server and a web browser. SSL works by ensuring that any data transferred between users and websites, or between two systems, remains impossible to read. It uses encryption algorithms to scramble data in transit, which prevents hackers from reading it as it is sent over the connection. This data includes potentially sensitive information such as names, addresses, credit card numbers, or other financial details.

The authentication process works like this:

1. The user enters the URL in the browser.
2. The client's request for the web page is sent to the server.
3. The server receives the request and sends its server certificate to the client.
4. The client's browser checks its certificate store for a certificate from the CA that issued the server certificate.
5. If the CA certificate is found, the browser validates the certificate by checking the signature on the server's certificate using the public key provided on the CA's certificate.
6. If this test is successful, the browser accepts the server certificate as valid.
7. A symmetric encryption key is generated and encrypted by the client, using the server's public key.
8. The encrypted key is returned to the server.
9. The server decrypts the key with the server's own private key. The two computers now share an encryption key that can be used to secure communications between them.

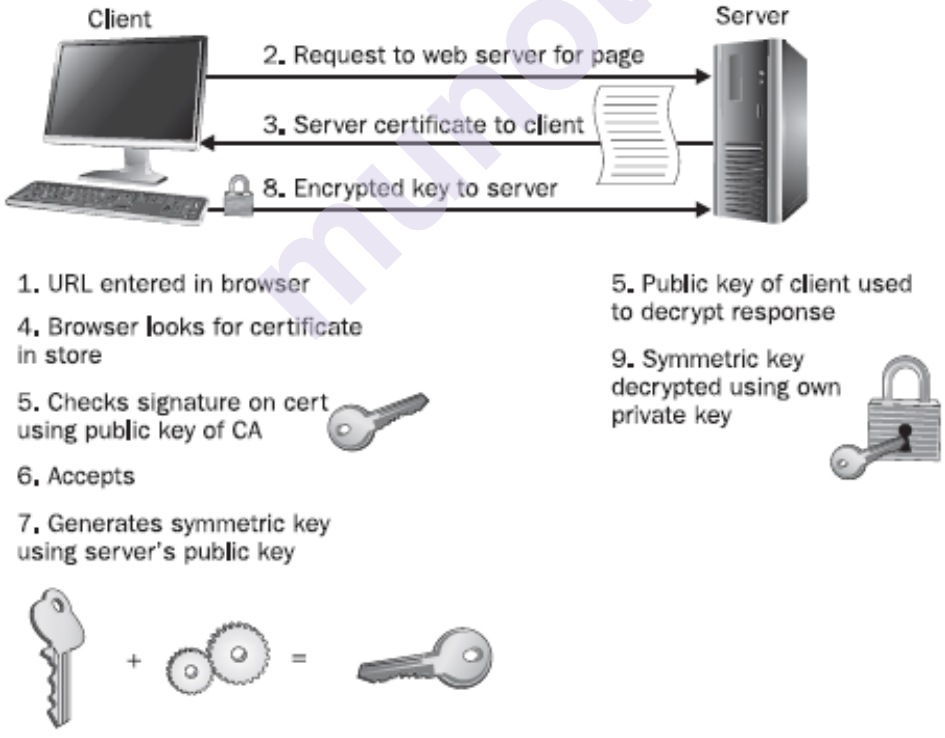


Figure 3.5 SSL for secure communications between a web server and a client

3.2.4 Biometrics:

In biometric authentication, **"something you have"** is something that is physically part of you. Biometric systems include the use of facial recognition and identification, retinas, iris scans, fingerprints, hand geometry, voice recognition, lip movement, and keystroke analysis. Biometric devices are used for security identification and authentication. These devices can recognize a user and then correctly prove whether the identified user holds the identity they claim to have. Biometric security systems use automated techniques, in which human intervention is reduced to the minimum to recognize and then confirm an individual's identity based on distinctive physiological or behavioral features, such as fingerprints, face pictures, iris recognition, and voice recognition.

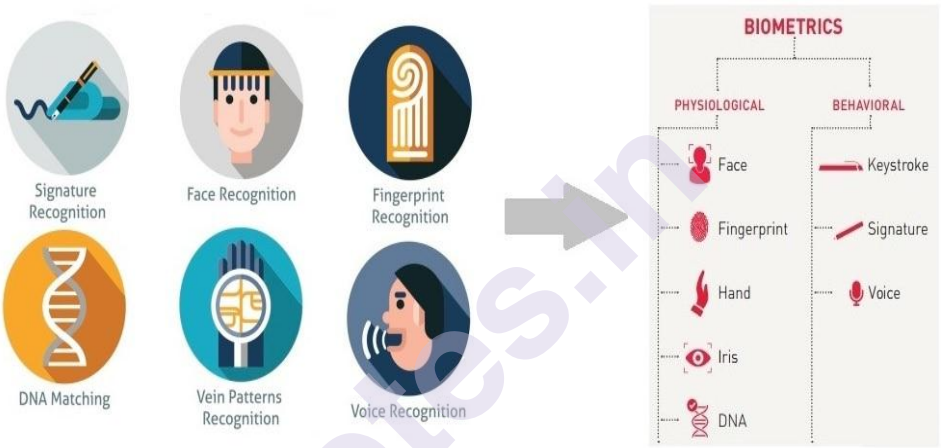


Figure 3.6 Biometric authentication

The process hinges on two things: first, that the body part examined can be said to be unique, and second, that the system can be tuned to require enough information to establish a unique identity and not result in a false rejection, while not requiring so little information as to provide false positives. All of the biometrics currently in use have been established because they represent characteristics that are unique to individuals. The relative accuracy of each system is judged by the number of false rejections and false positives that it generates.

In addition to false negatives and false positives, biometrics live under the shadow, popularized by the entertainment industry, of malicious attackers cutting body parts from the real person and using them to authenticate to systems.

Other attacks on fingerprint systems have also been demonstrated—one such is the **gummy finger attack**. In May 2002, Tsutomu Matsumoto, a graduate student of environment and information science at Yokohama National University, obtained an imprint of an audience member's finger and prepared a fake finger with the impression. He used about $10 of commonly available items to produce something the texture of the candy gummy worms. He then used the "gummy finger" to defeat ten different commercial fingerprint readers. While this attack would require access to

the individual's finger, another similar attack was demonstrated in which Matsumoto used latent fingerprints from various surfaces. This attack was also successful. These attacks not only defeat systems most people believe to be undefeatable, but after the attack, you can eat the evidence!

3.3 AUTHORIZATION

Authentication is an act of validating who the user is; authorization specifies what that user can do. Typically thought of as a way of establishing access to resources, such as files and printers, authorization also addresses the suite of privileges that a user may have on the system or on the network. In its ultimate use, authorization even specifies whether the user can access the system at all.

Let us take the example of boarding a plane. You have your boarding pass that states you are authorized to fly with that plane. However, it is not enough for the gate agent to let you get on board. You also need your passport stating your identity. In this case, the gate agent compares the name on the passport with the name on the boarding pass and lets you go through it if they match. In the authorization context, your name is an **attribute** of your identity. Other attributes are your age, your language, your credit card, and anything else relevant in a specific case. Your name on the passport is a **claim**, that is, a declaration stating you've got that attribute. Someone reading the name on your passport can be sure of your name because they trust the government which issued your passport. The boarding pass along with the proof of identity of consumers represents a kind of 'access token' that grants access rights to jump onto the plane. In the scenarios described above, you can see that the act of authorizing enables entities to execute tasks that other entities are not allowed to complete. Computer systems that use authorization work in a similar manner.

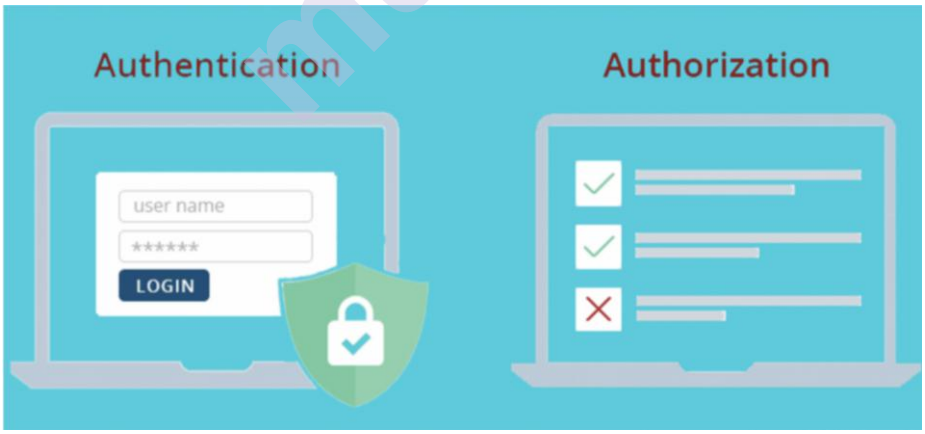


Figure 3.7 Authorization

There are a variety of types of authorization systems, including user rights, role-based authorization, access control lists, and rule-based authorization.

3.3.1 User Rights:

Privileges or user rights are different from permissions. User rights provide the authorization to do things that affect the entire system. The ability to create groups, assign users to groups, log in to a system, and many more user rights can be assigned. Other user rights are implicit and are rights that are granted to default groups—groups that are created by the operating system instead of by administrators. These rights cannot be removed.

3.3.2 Role-Based Authorization (RBAC):

Each job within a company has a role to play. Each employee requires privileges (the right to do something) and permissions (the right to access particular resources and do specified things with it) to do their job. Early designers of computer systems recognized that the needs of possible users of systems would vary, and not all users should be given the right to administer the system.

Two early roles for computer systems were user and administrator. Early systems defined roles for these types of users to play and granted them access based on their membership in one of these two groups. Administrators (superusers, root, admins, and the like) were granted special privileges and allowed access to a larger array of computer resources than that of ordinary users. Administrators, for example, could add users, assign passwords, access system files and programs and reboot the machine. Ordinary users could log in and perhaps read data, modify it, and execute programs. This grouping was later extended to include the role of auditor (a user who can read system information and information about the activities of others on the system, but not modify system data or perform other administrator role functions).

As systems grew, the roles of users were made more granular. Users might be quantified by their security clearance, for example, and allowed access to specified data or allowed to run certain applications. Other distinctions might be made based on the user's role in a database or other application system. Commonly, roles are assigned by departments such as Finance, Human Resources, Information Technology, and Sales.

3.3.3 Access Control Lists (ACLs):

Attendance at some social events is limited to invitees only. To ensure that only invited guests are welcomed to the party, a list of authorized individuals may be given to those who permit the guests in. If you arrive at the event, the name you provide is checked against this list and then an entry is granted or denied. Authentication, in the form of a photo identification check, may or may not play a part here, but this is a good, simple example of the use of an access control list (ACL).

Information systems may also use ACLs to determine whether the requested service or resource is authorized. Access to files on a server is often controlled by information that is maintained on each file. Likewise,

the ability for different types of communication to pass a network device can be controlled by ACLs.

3.3.4 Rule-Based Authorization:

Rule-based authorization requires the development of rules that stipulate what a specific user can do on a system. These rules might provide information such as "User Alice can access resource Z but cannot access resource D." More complex rules specify combinations, such as "User Bob can read file X only if he is sitting at the console in the data center." In a small system, rule-based authorization may not be too difficult to maintain, but in larger systems and networks, it is excruciatingly tedious and difficult to administer.

3.4 ENCRYPTION

Encryption is a way of scrambling data so that only authorized parties can understand the information is an ancient practice. It evolved into the modern practice of cryptography—the science of secret writing, or the study of obscuring data using algorithms and secret keys.

History of Encryption:

Once upon a time, keeping data secret was not hard. Hundreds of years ago, when few people were literate, the use of written language alone often sufficed to keep information from becoming general knowledge. To keep secrets then, you simply had to write them down, keep them hidden from those few people who could read, and prevent others from learning how to read. Deciphering the meaning of a document is difficult if it is written in a language you do not know.

Early Codes:

Early code used transposition. They simply rearranged the order of the letters in a given message. This rearrangement had to follow some order, otherwise, the recipient would not be able to restore the original message. The use of the scytale by the Spartans in the fifth-century b.c. is the earliest record of a pattern being used for a transposition code. The scytale was a rod around which a strip of paper was wrapped. The message was written down the side of the rod, and when it was unwound, the message was unreadable. If the messenger was caught, the message was safe. If he arrived safely, the message was wound around an identical rod and read.

Other early attempts at cryptography (the science of data protection via encryption) used substitution. A substitution algorithm replaces each character in a message with another character. Caesar's cipher is an example of a substitution algorithm. It is a type of substitution algorithm in which each letter in the plaintext is 'shifted' a certain number of places down the alphabet. For example, with a shift of 1, A would be replaced by B, B would become C, and so on.

Example:

To pass an encrypted message from one person to another, it is first necessary that both sender and receiver have the 'key' for the cipher, so that the sender may encrypt it and the receiver may decrypt it. For the Caesar cipher, the key is the number of characters to shift the cipher alphabet.

Here is an example of the encryption and decryption steps involved with the Caesar cipher. The text we will encrypt is 'my password is root', with a shift (key) of 1.

plain text: my password is root

cipher text: nz qbttxpse jt sppu

It is easy to see how each character in the plaintext is shifted up the alphabet. Decryption is just as easy, by using an offset of -1.

plain text: abcdefghijklmnopqrstuvwxyz

cipher text: bcdefghijklmnopqrstuvwxyza

Obviously, if a different key is used, the cipher alphabet will be shifted a different amount.

The use of such codes, in which knowledge of the algorithm is all that keeps the message safe, has long been known to be poor practice. Sooner or later, someone will deduce the algorithm, and all is lost.

3.4.1 Symmetric-Key Cryptography:

Symmetric key cryptography is a type of encryption in which a similar key is used to encrypt and decrypt messages. This secret key is known only to the sender and to the receiver. It is also called **secret-key cryptography**. The message exchange using symmetric key cryptography involves the following steps-

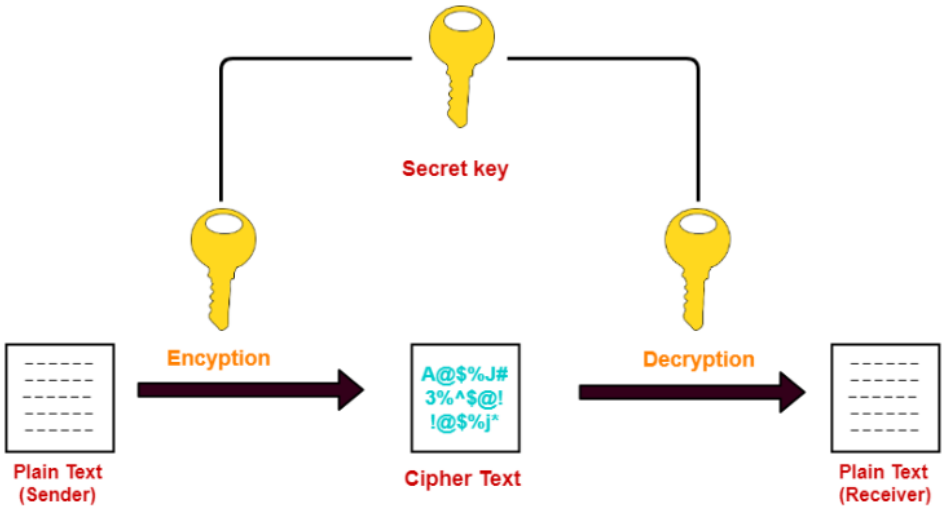


Figure 3.8 Symmetric Key Cryptography

Before starting the communication, sender and receiver share the secret key. This secret key is shared through some external means. At sender side, sender encrypts the message using his copy of the secret key. The cipher text is then sent to the receiver over the communication channel. At receiver side, receiver decrypts the cipher text using his copy of the secret key. After decryption, the message converts back into readable format.

Some of the encryption algorithms that use symmetric key are:

- Advanced Encryption Standard (AES)
- Data Encryption Standard (DES)

For example, suppose we take a plaintext message, "hello," and encrypt it with a key; let's say the key is "2jd8932kd9." Encrypted with this key, our simple "hello" now reads "X5xJCSycg15=", which seems like random garbage data. However, by decrypting it with that same key, we get "hello" back.

Plaintext + key = ciphertext:

hello + 2jd8932kd9 = X5xJCSycg14=

Ciphertext + key = plaintext:

X5xJCSycg15= + 2jd8932kd9 = hello

(This is an example of symmetric encryption, in which only one key is used.)

The advantages of symmetric key algorithms are:

- They are efficient.
- They take less time to encrypt and decrypt the message.

3.4.2 Public Key Cryptography:

Public key encryption, or public key cryptography, is a method of encrypting data with two different keys and making one of the keys, the public key, available for anyone to use. The other key is known as the private key. Data encrypted with the public key can only be decrypted with the private key, and data encrypted with the private key can only be decrypted with the public key. Public key encryption is also known as asymmetric key cryptography.

The message exchange using public key cryptography involves the following steps:

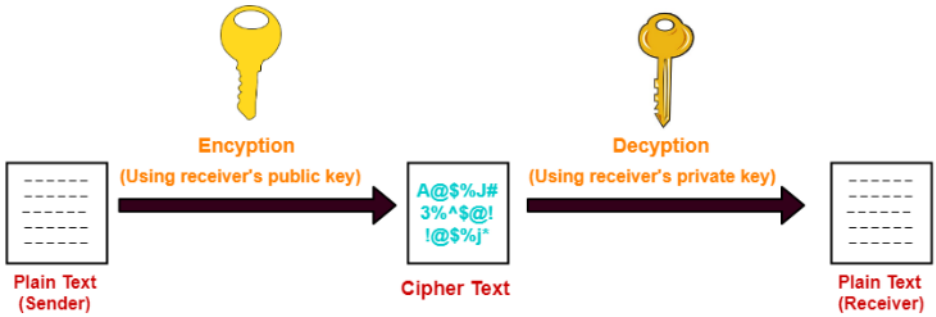


Figure 3.9 Public Key Cryptography

The famous asymmetric encryption algorithms are:

1. RSA Algorithm
2. Diffie-Hellman Key Exchange

3.5 PUBLIC KEY INFRASTRUCTURE

Public Key Infrastructure (PKI) has become one of the most prevalent forms of encryption in modern electronic transactions. Today, organizations rely on PKI to manage security through encryption. The most common form of encryption used today involves a public key, which anyone can use to encrypt a message, and a private key (also known as a secret key), which only one person should be able to use to decrypt those messages. These keys can be used by people, devices, and applications. An associated key pair is bound to a security principal (user or computer) by a certificate. A certificate authority (CA) issues, catalogs, renew, and revokes certificates under the management of a policy and administrative control. Common examples of PKI security today are SSL certificates on websites so that site visitors know they're sending information to the intended recipient, digital signatures, and authentication for Internet of Things devices.

There are three key components: digital certificates, certificate authority, and registration authority.

1. Digital Certificates:

PKI functions because of digital certificates. A digital certificate is like a driver's license—it's a form of electronic identification for websites and organizations. Secure connections between two communicating machines are made available through PKI because the identities of the two parties can be verified by way of certificates. So how do devices get these certificates? You can create your own certificates for internal communications. If you would like certificates for a commercial site or something of a larger scale, you can obtain a PKI digital certificate through a trusted third-party issuer, called a Certificate Authority.

2. Certificate Authority:

A Certificate Authority (CA) is used to authenticate the digital identities of the users, which can range from individuals to computer systems to servers. Certificate Authorities prevent falsified entities and manage the life cycle of any given number of digital certificates within the system. Much like the state government issuing you a license, certificate authorities vet the organizations seeking certificates and issue one based on their findings. Just as someone trusts the validity of your license based on the authority of the government, devices trust digital certificates based on the authority of the issuing certificate authorities. This process is similar to how code signing works to verify programs and downloads.

3. Registration Authority:

Registration Authority (RA), which is authorized by the Certificate Authority to provide digital certificates to users on a case-by-case basis. All of the certificates that are requested, received, and revoked by both the Certificate Authority and the Registration Authority are stored in an encrypted certificate database.

Certificate history and information is also kept on what is called a certificate store, which is usually grounded on a specific computer and acts as a storage space for all memory relevant to the certificate history including issued certificates and private encryption keys. Google Wallet is a great example of this.

3.6 SUMMARY

Authentication is the process of proving you are who you say you are. If someone possesses your user credentials, it is possible for that person to say they are you, and to prove it to the satisfaction of the system. While many modern systems are based on hardware, such as tokens and smart cards, and on processes that can be assumed to be more secure, such as one-time passwords, most systems still rely on passwords for authentication. These systems are not well protected because passwords are a terrible way to identify people. Other authentication methods are better. You should always evaluate an authentication system based on how easy it would be to defeat its controls.

Authorization, on the other hand, determines what an authenticated user can do on the system or network. A number of controls exist that can help define these rights of access explicitly. User rights are often provided directly by the operating system, either via permissions granted to the user account directly or through the use of groups. If the user account belongs to a particular group, it is granted rights to do certain things. This method of authorization, while commonly found in most organizations, is not easy to manage and has a high potential for error. Role-based access controls are similar to group authorization, but they are organized into sets of functions based on some key common characteristic.

In this chapter, we started with a brief history of encryption in order to establish a context regarding the limited lifespan of cryptographic techniques. Looking at early codes, and the progression to more modern codes, we saw how the encryption methods evolve to stay one step ahead of those who want to break the confidentiality of the protected data. Symmetric-key cryptography evolved naturally from early methods of hiding data using mathematical transformations. In these algorithms, key exchange is a key challenge. Whoever possesses the key can decrypt the message—thus, properly secured key exchange is critical to the continued confidentiality of the data. Public key cryptography is the next evolution of encryption. Using two keys, one public and one private, helps deal with the problem of key exchange that was encountered in symmetric-key encryption. Public Key Infrastructure (PKI) uses public key cryptography to create certificates, which are used for a variety of purposes.

3.7 QUESTIONS

1) Explain Authentication.
2) Describe Biometric authentication.
3) Explain Authorization.
4) Write short note on Encryption.
5) Explain Symmetric Key Cryptography.
6) Explain Public Key Cryptography.
7) Explain Certificate Based authentication.

3.8 REFERENCES

- https://docs.oracle.com/cd/E19424-01/820-4811/gdzeq/index.html
- https://www.cm.com/en-in/glossary/what-is-one-time-password/
- https://www.thalesgroup.com/en/markets/digital-identity-and security/government/inspired/biometrics
- https://web.mit.edu/6.857/OldStuff/Fall03/ref/gummy-slides.pdf
- https://www.gatevidyalay.com/cryptography-symmetric-key-cryptography/
- The Complete Reference: Information Security, Mark Rhodes-Ousley, McGrawHill, Second Edition.

*****

4

STORAGE SECURITY, DATABASE SECURITY

Unit Structure

4.0 OBJECTIVES

- Learn Modern Storage Security
- Learn Database Security and Database Security Layers
- Database Backup and Recovery
- Database Auditing and Monitoring

4.1 INTRODUCTION

The primary concern of network security is to protect assets (data) that reside on the network. Data resides in storage, which is either controlled or unmanaged. Storage technologies have evolved over the past decade in complexity, capability, and capacity. The effectiveness of storage security controls and technologies has advanced accordingly. Today's storage technologies can protect data natively in many ways; for example, many modern storage technologies have built-in encryption and access control to protect confidentiality, integrity, and redundancy to protect the availability and onboard protection against malware. In this chapter, we'll cover the ways in which the built-in security features of modern storage infrastructures can be leveraged to protect data. We'll review best practices for building storage infrastructures to provide the best protection for data assets. Let's begin with a look at how storage security has changed in recent years.

4.1.1 Evolution of Storage Security:

Almost ten years ago, 3.5-inch floppy disk drives were still included on some computers. Being portable storage devices, floppy disks were hard to secure. They were easily lost or the data on them became corrupted. They could be used to propagate malware. The use of floppy disks was largely phased out by the late 2000s.

The next generation of storage devices, compact discs (CDs) and digital video discs (DVDs), posed a unique threat due to their longevity. Unlike other, more volatile storage media, these polycarbonate-encased metal optical data storage devices seem like they will last forever if handled properly. If someone places private, confidential data on a CD or DVD and then misplaces the disc, who knows how long it might stick around and who may discover it in the future? For this reason, optical storage devices were banned in many corporate environments, especially those required to comply with privacy regulations. Moreover, once the data is burned to the media, it can't be changed, so you can't retroactively apply protection to it.

Flash drives (USB sticks and the like) have become popular over the past few years. These devices have become so cheap and prevalent that they have practically supplanted optical storage devices. Who needs to burn when you can simply copy? Flash drives are a significant source of malware infections in many environments. In addition, they make data theft remarkably easy with their small size, portability, and compatibility with every major computing platform.

Portable hard drives, like flash drives, are cheap and plentiful. Portable USB hard drives have so much storage capacity that they can be used to steal all the data in many organizations. Even modern smartphones, cameras, and tablets contain large amounts of flash memory and are accessible via USB, allowing data thieves to copy files unobtrusively.

In addition to the previously mentioned dedicated storage devices, the security practitioner now also has to contend with smartphones and mobile devices, which have significant amounts of onboard storage. These devices pose a significant risk to an organization's data because they are less "obvious" than a hard drive or memory stick and because any stolen data hiding on them can be hard to detect.

All of the storage devices mentioned so far are considered to be unmanaged. The best protections for them are encryption and access control. Encrypting confidential data can stop, or discourage, data theft. Information rights management can protect confidential documents such that even if they are stolen, they can't be opened by unauthorized users. In addition, USB device control software can block access to the USB ports on computers where it's installed, and it can allow or block various activities such as copying to or from USB devices, based on the type of document. Ultimately, unmanaged storage devices are hard to secure and hard to control. That's why organizations have turned to managed storage, which allows their data to be accessed in secure, controlled ways. With managed storage, organizations can block USB storage devices and drive users toward the managed storage instead.

4.2 MODERN STORAGE SECURITY

Modern storage solutions have moved away from endpoint computers to the network. Network-attached storage (NAS) and storage area networks (SANs) consist of large hard drive arrays with a controller that serves up their contents on the network. NAS can be accessed by most computers and other devices on the network, while a SAN is typically used by servers. These storage systems have many built-in security features to choose from. Based on the security requirements of the environment, these security settings can be configured to meet the objectives of the security policy. Modern storage environments can be considered separate IT infrastructures of their own. Many organizations are now dividing their IT organizations along the lines of networks, servers, and, storage—acknowledging that storage merits a place alongside these long-venerated institutions.

4.2.1 Storage Infrastructure:

Storage infrastructure refers to the overall set of hardware and software components needed to facilitate storage for a system. This is often applied to cloud computing, where cloud storage infrastructure is composed of hardware elements like servers, as well as software elements like operating systems and proprietary delivery applications.

Cloud storage infrastructure and other types of storage infrastructure can vary quite a bit, partly because of new and emerging storage technologies. For example, with storage virtualization, the infrastructure is changed to become more software-driven than hardware-driven. In a typical storage virtualization environment, a set of physical hard drives are replaced by a set of "logical drives" or "virtual drives" that are partitioned and operated by software. Engineers use different types of strategies like a redundant array of independent disks (RAID) design to create more versatile storage systems that use hardware in more sophisticated ways.

4.3 RISKS TO DATA

The first risk involves data that can be accessed via an unauthorized system. The second risk is data access by unauthorized persons.

Risk Remediation:

In this section, we have categorized the risks associated with data storage according to the classic CIA triad of Confidentiality, Integrity, and Availability. For each identified risk, where possible, security controls consistent with the “three Ds” of security—defense, detection, and deterrence—are applied in an effort to mitigate the risk using the principle of layered security (also known as defense-in-depth). What’s left after those controls are applied to mitigate the risks is then identified as residual risks.

4.3.1 Confidentiality Risks:

Confidentiality risks are associated with vulnerabilities and threats pertaining to the privacy and control of information, given that we want to make the information available in a controlled fashion to those who need it without exposing it to unauthorized parties.

4.3.1.1 Data Leakage, Theft, Exposure, Forwarding:

A data leak is when sensitive data is accidentally exposed physically, on the Internet, or in any other form including lost hard drives or laptops. This means a cybercriminal can gain unauthorized access to sensitive data without effort. There are four major threat vectors for data leakage: theft by outsiders, malicious sabotage by insiders (including unauthorized data printing, copying, or forwarding), inadvertent misuse by authorized users, and mistakes created by unclear policies.

Defense:

Employ software controls to block inappropriate data access using a data loss prevention (DLP) solution and/or an information rights management (IRM) solution.

Detection:

Use watermarking and data classification labeling techniques along with monitoring software to track the flow of data.

Deterrence:

Establish security policies that assign serious consequences to employees who leak data, and include clear language in contracts with service providers specifying how data privacy is to be protected and maintained, and what penalties will be there for failure to protect and maintain it.

Residual risks:

Data persistence within the storage environment can expose data after it is no longer needed, especially if the storage is hosted on a vendor-provided service that dynamically moves data around in an untraceable manner. Administrative access that allows system administrators full access to all files, folders, and directories, as well as the underlying storage infrastructure itself, can expose private data to administrators.

4.3.1.2 Espionage, Packet Sniffing, Packet Replay:

Espionage refers to the unauthorized interception of network traffic to gain information intentionally. Packet sniffing is the act of gathering, collecting, and monitoring the data packets that travel through a computer network or the internet. Using tools to reproduce traffic and data that was previously sent on a network is called packet replay.

Defense:

Encrypt data at rest and in transit through the use of modern robust encryption technologies for file encryption, and network encryption between servers and over the Internet.

Detection:

An information rights management (IRM) solution can keep track of data access, which can provide the ability to detect unauthorized access attempts. In addition, an intrusion detection system (IDS) can help identify anomalous behavior on the network that may indicate unauthorized access.

Deterrence:

In storage environments that are hosted by a third party, employ contract language that makes the service provider liable for damages resulting from unauthorized access.

Residual risk:

Data can be stolen from the network through tools that take advantage of network topologies, network weaknesses, compromised servers, network equipment, and direct access to network devices.

4.3.1.3 Inappropriate Administrator Access:

If users are given privilege levels usually reserved for system administrators which provide full access to a system and all data that the system has access to, they will be able to view data or make changes without being properly restricted through the system's authorization processes. Administrators have the authority to bypass all security controls, and they can be used to intentionally or mistakenly compromise private data.

Defense:

Reduce the number of administrators for each function (servers, network, and storage) to as low a number as possible (definitely fewer than ten and preferably fewer than five) and ensure that background checks are used to screen personnel who have administrative access. A vendor security review should be performed to validate these practices before engaging any vendors.

Detection:

Review the provider's administrative access logs for its internal infrastructure on a monthly or quarterly basis. Review the provider's list of administrators on a biannual basis.

Deterrence:

Establish security policies, especially for administrators that assign serious consequences for inappropriate data access. In hosted environments, select only providers that have good system and network administration practices and make sure their practices are reviewed regularly.

Residual risk:

Because administrators have full control (all rights), they can abuse their access privileges either intentionally or accidentally, resulting in the compromise of personal information or service availability.

4.3.1.4 Storage Persistence:

Data remains on storage devices long after it is no longer needed, and even after it is deleted. Data that remains in storage after it is no longer needed, or that is deleted but not strongly overwritten, poses a risk of later discovery by unauthorized individuals.

Defense:

Maintain a U.S. Department of Defense (DoD) level program of disk wiping or file shredding when disks are decommissioned or replaced, and after old data is archived.

Detection:

There isn't much that can be done to discover that your data persists on a disk that has been taken offline.

Deterrence:

Establish data-wiping requirements before selecting a storage product and ensure that contract language establishes these requirements.

Residual risk:

Data can remain on physical media long after it is thought to have been deleted. Later data can be recovered.

4.3.1.5 Misuse of Data:

People who have authorized access to data can misuse the data that they are not supposed to do. Examples are employees who leak information to competitors, developers who perform testing with production data, and employees who take data out of the controlled environment of the organization's network into their unprotected home environment.

Defense:

For employees, use security controls similar to those in private data networks, such as DLP, RBAC, and scrambling of test and development data. Block the ability to send e-mail attachments to external e-mail addresses.

Detection:

Use watermarking and data classification labeling along with monitoring software to track data flow. IRM can be used to perform these functions.

Deterrence:

Employ a strict security policy paired with an awareness program to deter people from extracting data from controlled environments and moving it to uncontrolled environments.

Residual risk:

People can find ways around controls and transfer data into uncontrolled environments, where it can be stolen or misused.

4.3.1.6 Fraud:

A person who illegally or deceptively gains access to information they are not authorized to access commits fraud. Fraud may be perpetrated by outsiders but is usually committed by trusted employees.

Defense:

Use checks and balances along with separation of duties and approvals to reduce the dependence on single individuals for information access, so if somebody does perform a fraudulent action, it will be noticed. This can also be a deterrent action.

Detection:

Perform regular audits on computing system access and data usage giving special attention to unauthorized access.

Deterrence:

Ensure that security policies include penalties for employees who access data they are not authorized for. In hosted environments, transfer risk to service providers using contractual language that holds the service provider responsible for fraud committed by a service provider employee.

Residual risk:

Fraudulent data access can occur despite the controls that are designed to prevent it.

4.3.1.7 Hijacking:

Hijacking in the context of computing refers to the exploitation of a valid computer session also called a session key to gain unauthorized access to information or services in a computer system. In particular, it's the theft of a magic cookie used to authenticate a user to a remote server. For example, the HTTP cookies used to maintain a session on many websites can be stolen using an intermediary computer or with access to the saved cookies on the victim's computer. If an attacker can steal the authentication cookie, they can make requests themselves as if they were genuine users, gaining access to privileged information or they may modify data. If this cookie is a persistent cookie, then the impersonation can continue for a considerable period. Any protocol in which a state is maintained using a key passed between two parties is vulnerable, especially if it's not encrypted.

Defense:

Look for solid identity management solutions that specifically address this risk using strong, difficult-to-guess session keys with encryption. Use good key management, key escrow, and key recovery practices as a customer so that employee departures do not result in the inability to manage your data.

Detection:

Routinely monitor logs, looking for unexpected behavior.

Deterrence:

Not much can be done to deter attackers from hijacking sessions, other than an aggressive legal response.

Residual risk:

Attackers can impersonate valid users or even use administrative credentials to lock you out or damage your infrastructure.

4.3.1.8 Phishing:

Phishing attacks are the practice of sending fraudulent communications that appear to come from a reputable source. Phishing is an attempt to trick a victim into disclosing personal information. The most common method of phishing is to send potential victims an e-mail message that appears to be from a legitimate organization and directs the recipients to log in and provide a username, password, credit card information, or other sensitive information.

Defense:

Employ anti-phishing technologies to block rogue websites and detect false URLs. Use multifactor authentication for customer-facing systems to ensure that users are aware when they are redirected to a fake website. Send periodic informational updates and educational materials to customers explaining how the system works and how to avoid phishing attempts. Never send e-mails that include or request personal details, including ID or passwords.

Detection:

Use an application firewall to detect when remote websites are trying to copy or emulate your website.

Deterrence:

Maintain educational and awareness programs for individuals who use and store personal information of employees or customers.

Residual risk:

Employees can become victims of phishing scams despite the best training and awareness programs, especially if those scams are sophisticated. This can result in data loss.

4.4 INTEGRITY RISKS

Integrity risks affect both the validity of information and the assurance that the information is correct. Some government regulations are

particularly concerned with ensuring that data is accurate. If information can be changed without warning, authorization, or an audit trail, its integrity cannot be guaranteed.

4.4.1 Malfunctions:

Computer and storage failures that corrupt data damage the integrity of that data.

Defense:

Make sure the storage infrastructure you select has appropriate RAID redundancy built in and that archives of important data are part of the service.

Detection:

Employ integrity verification software that uses checksums or other means of data verification.

Deterrence:

Due to the nature of data, because there is no human involvement, there isn't much that can be done.

Residual risk:

Technology failures that damage data may result in operational or compliance risk (especially relating to Sarbanes-Oxley requirements for publicly traded companies to ensure the integrity of their financial data).

4.4.2 Data Deletion and Data Loss:

Data can be accidentally or intentionally destroyed due to computer system failures or mishandling. Such data may include financial, organizational, personal, and audit trail information.

Defense:

Ensure that your critical data is stored and housed in more than one location (backup).

Detection:

Maintain and review audit logs of data deletion.

Deterrence:

Maintain educational and awareness programs for individuals who access and manage data. Ensure that data owners are assigned authority and control over data and responsibility for its loss.

Residual risk:

If important data is gone, it can't be restored under any circumstances.

4.4.3 Data Corruption and Data Tampering:

Changes to data caused by a malfunctioning in computer or storage systems, or by malicious individuals or malware, can damage the integrity of that data. Integrity can also be damaged by people who modify data with the intent to defraud.

Defense:

Utilize version control software to maintain archive copies of important data before it is modified. Ensure that all data is protected by antivirus software. Maintain role-based access control over all data based on least privilege principles, pursuant to job function and need to know.

Detection:

Use integrity-checking software to monitor and report alterations to key data.

Deterrence:

Maintain educational and awareness programs for individuals who access and manage data. Ensure that data owners are assigned authority and control over data and responsibility for its loss.

Residual risk:

Corrupted or damaged data can cause significant issues because valid, reliable data is the cornerstone of any computing system.

4.4.4 Accidental Modification:

This is the most common cause of data integrity loss, accidental modification occurs either when a user intentionally makes changes to data but makes the changes to the wrong data or when a user inputs data incorrectly.

Defense:

Utilize version control software to maintain archive copies of important data before it is modified. Maintain role-based access control over all data based on least privilege principles, pursuant to job function and need to know.

Detection:

Use integrity-checking software to monitor and report alterations to key data.

Deterrence:

Maintain educational and awareness programs for individuals who access and manage data. Ensure that data owners are assigned that have authority and control over data and responsibility for its loss.

Residual risk:

Corrupted or damaged data can cause significant issues because valid, reliable data is the cornerstone of any computing system.

4.5 AVAILABILITY RISKS

Availability risks are associated with vulnerabilities and threats pertaining to the reliability of services, given that we want the services that we use to be reliable, to pose a low risk, and to have a low incidence of an outage.

4.5.1 Denial of Service:

A denial-of-service (DoS) attack is a type of cyber attack in which a malicious actor aims to render a computer or other device unavailable to its intended users by interrupting the device's normal functioning. This type of attack commonly involves saturating the target machine with too many communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable.

Defense:

Select a storage platform that has solid protection against network attacks. Implement firewalls, an IPS, and network filtering at the perimeter of the storage network to block attacks.

Detection:

Monitor intrusion detection systems 24×7×365.

Deterrence:

Work with your legal department to ensure that attackers are found and prosecuted.

Residual risk:

Because most DoS and DDoS attacks make use of compromised systems across the globe, they can be hard to track, and because they flood system and network resources, they can get through an environment’s defenses.

4.5.2 Outage:

An outage is any unexpected downtime or unreachability of a computer system or network.

Defense:

The primary defense against any service outage is redundancy. Ensure that individual systems, devices, and network links are clustered or set up to use high availability. Outages are expensive—calculate the cost of downtime and use that to justify investment in the additional equipment needed for redundancy. Additionally, employ a solid disaster recovery plan to ensure that you are ready for extended outages so that the storage

environment can be automatically switched to a different location during an outage.

Detection:

Employ monitoring tools to continuously monitor the availability and response time of the storage environment.

Deterrence:

Because outages generally occur as a result of software problems, little can be done to stop them from happening.

Residual risk:

Unforeseen outages can occur even when all devices and network paths are completely redundant, due to malfunctions or human error, so storage infrastructures may be down for as long as it takes to switch over to the disaster recovery environment.

4.5.3 Instability and Application Failure:

Problems, such as flaws in software or firmware can cause freezing, locking, or crashing of applications making them unresponsive and resulting in loss of functionality or failure of an entire computer or network.

Defense:

Ensure that all software updates are applied to the infrastructure on a frequent basis.

Detection:

Implement service monitoring to detect and alert when an application does not respond correctly.

Deterrence:

In contracts with storage suppliers, include clear language that specifies penalties and remuneration for instability issues.

Residual risk:

Because instability in applications and infrastructure generally occurs as a result of software problems, little can be done to stop them from happening.

4.5.4 Slowness:

When the response time of a computer or network is considered unacceptably slow, its availability is affected.

Defense:

Using redundant storage systems and network connections set up the architecture so that application access will automatically switch to the fastest environment. Also, ensure that you have implemented high-capacity services with demand-driven expansion of resources.

Detection:

Monitor the response time of applications on a continuous basis and ensure that alerts have an out-of-band path to support staff so that response problems don't stop alerts from being delivered.

Deterrence:

Establish contract language with storage a manufacturer that provides compensation for unacceptable response times.

Residual risk:

Slowness can persist despite best efforts, resulting in a loss of efficiency and effective downtime.

4.5.5 High Availability Failure:

A service that is supposed to fail over in the event of a problem with one device to another, functioning devices may not actually fail over properly. This can happen, for example, when a primary device slows down to the point where it becomes effectively unresponsive, but the HA software doesn't actually consider it to be "down."

Defense:

Monitor the health of secondary systems or all systems in an HA cluster.

Detection:

Perform periodic failover testing.

Deterrence:

Not much can be done to guarantee that systems will switch over when they are supposed to.

Residual risk:

Sometimes, a primary device slows down to the point that it becomes unresponsive for all practical purposes, but because it's not officially "down" according to its software, the backup system doesn't take over.

4.5.6 Backup Failure:

When you discover that those backups you were relying on aren't actually any good, either because the media is damaged or the backup data is corrupted or missing, data is lost.

Defense:

Leverage storage elasticity to avoid the use of traditional offline (tape or optical) backups.

Detection:

Frequently perform recovery testing to validate the resilience of data.

Deterrence:

Establish a data-loss clause in the contract with the storage manufacturer so that they have the incentive to help with unforeseen loss of data.

Residual risk:

Backups fail, but multiple recovery paths can eliminate most of the risk. The practice of backing up data has been around for a long time and, consequently, is one of the most reliable security practices. As long as data is appropriately replicated, it can live forever, so the majority of residual risk, in this case, would be due to substandard data replication practices or lack of attention to this matter.

4.6 DATABASE SECURITY

Modern organizations rely heavily on the information stored in their database systems. From sales transactions to human resources records, mission-critical, sensitive data is tracked within these systems. It is very important that business and systems administrators take the proper precautions to ensure that these systems and applications are as secure as possible. You wouldn't want a junior-level database administrator to be able to access information that only the executive team should see, but you also wouldn't want to prevent your staff from doing their jobs. As with all security implementations, the key is to find a balance between security and usability.

Modern databases must meet different goals. They must be reliable, provide quick access to information and provide advanced features for data storage and analysis. Furthermore, they must be flexible enough to adapt to many different scenarios and types of usage. Many organizations rely on databases to serve as the "back end" for purchased applications or custom-developed applications. The "front end" of these systems is generally client applications or web user interfaces. Because of the heavy reliance that modern organizations place on their data storage systems, it's very important to understand, implement, and manage database security. Let's start by looking at an overview of various layers of database security and how they interact.

Database Security Layers:

Relational databases can support a wide array of different types of applications and usage patents; they generally utilize security at multiple layers. Each layer of security is designed for a specific purpose and can be

used to provide authorization rules. In order to get access to your most trusted information, users must have appropriate permissions at one or more of these layers.

4.6.1 Server-Level Security:

A database application is only as secure as the server it is running on. Therefore, it's important to start considering security settings at the level of the physical server or servers on which your databases will be hosted. In small simple configurations, you might need to secure only a single machine. Large organizations will likely have to make accommodations for many servers. These servers may be geographically distributed and even arranged in complex clustered configurations.

One of the first steps you need to take in order to secure a server is to determine which users and applications should have access to it. Modern database platforms are generally accessible over a network, and most database administration tasks can be performed remotely. It's also very important to physically protect databases in order to prevent unauthorized users from accessing database files and data backups. If an unauthorized user can get physical access to your servers, it's much more difficult to protect against further breaches.

4.6.2 Network-Level Security:

As mentioned previously, databases work with their respective operating system platforms to serve users with the data they need. Therefore, general operating system and network-level security also apply to databases. If the underlying platform is not secure, this can create significant vulnerabilities for the database. Since they are designed as network applications, you must take reasonable steps to ensure that only specific clients can access these machines.

Some standard "best practices" for securing databases include limiting the networks and/or network addresses that have direct access to the computer. For example, you might implement routing rules and packet filtering to ensure that only specific users on your internal network will even be able to communicate with a server.

As an example, Microsoft's SQL Server database platform uses a default TCP port of 1433 for communications between clients and the database. If you know for certain that there is no need for users on certain subnets of your network to be able to access this server directly, it would be advisable to block network access to this TCP port. Doing so can also prevent malicious users and code (such as viruses) from attacking this machine over the network. Another security practice involves changing the default port on which the server listens.

Of course, few real-world databases work alone. Generally, these systems are accessed directly by users, and often by mission-critical applications.

4.6.3 Operating System Security:

On most platforms, database security goes hand in hand with operating system security. Network configuration settings, file system permissions, authentication mechanisms, and operating system encryption features can all play a role in ensuring that databases remain secure. For example, on Windows-based operating systems, only the NTFS file system offers any level of file system security (FAT and FAT32 partitions do not provide any file system security at all). In environments that use a centralized directory services infrastructure, it's important for systems administrators to keep permissions settings up to date and to ensure that unnecessary accounts are deactivated as soon as possible. Fortunately, many modern relational database platforms can leverage the strengths of the operating systems that they run on. Let's look at this in more detail.

Managing Database Logins:

Most database systems require users to enter some authentication details before they can access a database. The first level of database security can be based on a standard username and password combination. Or, for improved manageability and single sign-on purposes, the database systems can be integrated with an organization's existing authentication system. For example, many relational database products that operate on Microsoft's Windows operating system platform can utilize the security features of a domain-based security model. Based on an individual's user account and group membership, he or she can perform a seamless "pass-through authentication" that does not require rekeying a username or password. Among the many benefits of this method is the ability to centrally administer user accounts. When a user account is disabled at the level of the organization's directory service, no further steps need to be taken to prevent the user from accessing database systems. In addition, organizations are increasingly turning to biometric-based authentication (authentication through the use of fingerprint identification, retinal scans, and related methods), as well as smart-card and token-based authentication. Database administrators can take advantage of these mechanisms by relying on the operating system for identifying users. Therefore, integrated security is highly recommended, both for ease of use and for ease of management.

Server logins can be granted permission directly. For example, a user may be given permission to shut down or restart a database or the ability to create a new database on the server. Login-level permissions generally apply to the server as a whole and can be used to perform tasks related to backup and recovery, performance monitoring, and the creation and deletion of databases. In some cases, users with server login permissions may be able to grant these permissions to other users. Therefore, it's very important to fully understand the security architecture of the database platform you're depending on to keep your information safe.

Another important consideration to keep in mind is that most relational database platforms allow operating system administrators to have much implicit permission on the database. For example, system administrators

can start and stop the services and can move or delete database files. Additionally, some database platforms automatically grant the system administrator a database login that allows full permissions. Although this is probably desirable in some cases, it's something that must be kept in mind when trying to enforce overall security. In some situations, it's important that not all system administrators have permission to access sensitive data that is stored on these servers. Configuring systems in this way can be a challenge, and the exact method of implementation will be based on the operating system and database platform you're running.

Most often, a server login only allows a user to connect to a database. It does not implicitly allow the user to perform any specific actions within databases. In the next section, we'll take a look at how database-level security can be used to assign granular permissions to database logins.

4.7 DATABASE BACKUP AND RECOVERY

An integral part of any overall database security strategy should be providing for database backup and recovery. Backups serve many different purposes. Most often, it seems that systems administrators perform backups to protect the information in the case of server hardware failures. Data can be lost due to accidental human errors, flawed application logic, defects in the database or operating system platform, and, of course, malicious users who are able to circumvent security measures. In the event, that data is incorrectly modified or destroyed altogether, the only method to recover data is from backups.

Since all relational database systems have some methods for performing database backups while a server is running, there isn't much of an excuse for not implementing backups. The real challenge is in determining what backup strategies apply to your environment. You'll need to find out what your working limitations are. This won't be an easy task, even in the best-managed organizations. It involves finding information from many different individuals and departments within your organization. You'll have to work hard to find existing data and make the best guesses and estimates for areas in which data isn't available.

So, how do you decide what to protect? One method is to classify the importance of the relative types of information you need to protect. For example, your sales databases might be of "mission critical" importance, whereas a small decision-support system might rank "low priority" on the scale (since the data can relatively easily be re-created, if necessary). It's also important to keep in mind that business managers may have a very different idea of the importance of data when compared to other users who actually deal with this information frequently. Keep in mind that determining how to protect information must be a team effort if it is to be accurate and successful. An example of high-level data protection requirements is shown in Table 4.1.

Resource	Importance	Notes
OLTP server	Critical	Information can't be easily re-created, and data loss will lead to inaccurate or misleading reports.
E-mail server	High	Recovering lost messages and user mailboxes is very difficult.
Decision-support server (data warehouse)	Medium	Information can be regenerated from other sources.
Intranet web server	Medium	Content is important, but is replicated among multiple machines as part of development processes.

Table 4.1 Sample Categorization of Data Based on Importance

4.7.1 Determining Backup Constraints:

Once you have an idea of what your organization needs to back up, it's time to think about ways in which you can implement a data protection strategy. It is of critical importance that you define your business requirements before you look at the technical requirements for any kind of data protection solution. Table 4.2 provides an example of a requirements worksheet that summarizes data protection needs.

In addition to these requirements, you might also have a preliminary budget limit that can serve as a guideline for evaluating solutions. You should also begin thinking about personnel and the types of expertise you'll need to have available to implement a solution.

Machine	Amount of Data (est.)	Backup Window	Acceptable Downtime	Acceptable Data Loss	Other Requirements
Server 1 (file/ print services)	14GB	>12 hours	1 day	1 day	General file/ print server
Server 2 (file services)	>17GB	>6 hours	3 hours	4 hours	Engineering file server
SQL Server 1 (sales OLTP)	>6GB	>12 hours	30 minutes	1 hour	Sales order entry; must support point-in-time recovery
Shipping server	>17.5GB	>2 hours	5 minutes	None	Must remain online at all times; transactions cannot be lost

Table 4.2 Sample Data Protection Requirements Worksheet Based on Business Requirements

4.7.2 Determining Recovery Requirements:

The purpose of data protection is not to create backups. The real purpose is to provide the ability to recover information, in case it is lost. To that end, a good practice is to begin designing a backup solution based on your recovery requirements. You should take into account the cost of downtime, the value of the data, and the amount of acceptable data loss in a worst-case scenario. Also, keep in mind the likelihood of certain types of disasters.

When planners are evaluating business needs, they may forget to factor in the potential time for recovering information. The question they should

ask is the following: "If we lose data due to failure or corruption, how long will it take to get it back?" In some cases, the answer will be based on the technical limitations of the hardware you select. For example, if you back up 13GB of data to tape media and then the database becomes corrupted, the recovery time might be two hours. But what if that's not fast enough? Suppose your systems must be available within half that time—one hour. In that case, you'll need to make some important decisions. An obvious choice is to find suitable backup hardware to meet these constraints. If budgetary considerations don't allow that, however, you'll need to find another way.

4.8 KEEPING YOUR SERVERS UP TO DATE

An important security best practice that also applies to databases is keeping systems up to date. In order to ensure that known vulnerabilities and server problems are repaired, you must apply the latest security and application patches. It's especially difficult to keep active databases up to date, since downtime, testing, and potential performance degradation can be real concerns. However, you should always check for available updates and find out if the servers you manage have problems that are potentially solved by an update. If so, plan to install the updates as soon as you can test and deploy them. Additionally, relevant patches should be applied to the operating system on which the database is running. Most database vendors offer support websites that offer technical details and updates for their server platforms.

4.9 DATABASE AUDITING AND MONITORING

The idea of accountability is an important one when it comes to network and database security. The process of auditing involves keeping a log of data modifications and permissions usage. Often, users that are attempting to overstep their security permissions (or users that are unauthorized altogether) can be detected and dealt with before significant damage is done; or, once data has been tampered with, auditing can provide details about the extent of loss or data changes. There's another benefit of implementing auditing: when users know that certain actions are being tracked, they might be less likely to attempt to snoop around your databases. Thus, this technique can serve as a deterrent. Unfortunately, in many environments, auditing is overlooked.

Though it won't necessarily prevent users from modifying information, auditing can be a very powerful security tool. Most relational databases provide you with the ability to track specific actions based on user roles or to track actions on specific database objects. For example, you might want to create an audit log entry whenever information in the EmployeeSalary table is updated, or you might choose to implement auditing of logins and certain actions to deter systems administrators (who might require full permissions on a database) from casually "snooping around" in a database.

Perhaps one of the reasons that auditing is not often implemented is that it requires significant planning and management. Unlike some types of "set

and forget" functions, it's important to strike a balance between technical requirements and capturing enough information to provide meaningful analysis. In many cases, auditing too much information can decrease system performance. Also, audit logs can take up significant disk space. Finally, a few database administrators would enjoy the task of looking through thousands of audit log entries just to find a few items that may be of interest.

Most relational database systems offer some level of auditing functionality. Even if one or more of the types of databases you support does not include this feature, you can always implement your own. At a minimum, most database administrators should configure logging of both successful and failed database login attempts. Although this measure, by itself, will provide limited information, it will provide for some level of accountability. Of course, capturing data is only one part of overall auditing.

4.10 SUMMARY

As the storage of data has evolved from individually carried media to a specialized infrastructure environment, storage now requires specific planning and implementation of security in order to protect the data. This chapter has presented several options, techniques, and best practices to equip the storage administrator to make the best choices for the specific environment of the organization.

In this chapter, we covered a lot of information that is specific to implementing and maintaining security for relational databases. Although many of the same policies, procedures, tools, and techniques covered in earlier chapters also apply to databases, there are some special considerations that should be kept in mind. We began by looking at the roles that databases can play in a typical organization. Then we examined the various levels of security that are implemented in most relational database platforms. Specifically, we looked at server-level, network-level, and database-level security. The permissions at each of these levels can help narrowly define what users can and cannot do, and can help prevent accidental or malicious data modifications. Next, we looked at how application-level security can be used to maintain strict permissions while simplifying database administration. Another important aspect related to ensuring the security of database systems is implementing a data protection plan. We looked at the reasons for performing backups, how backups should be planned, and various backup operations that can be performed in relational databases. Finally, we looked at the importance of auditing and monitoring servers.

4.11 QUESTIONS

1) Write a short note on Storage security evolution.

2) Explain risk remediation for integrity risk.

3) Explain risk remediation for confidentiality risk.

4) Explain risk remediation for availability risk.

5) Describe various Database Security layers.

6) Explain Database Backup and recovery.

7) What are database auditing and monitoring?

4.12 REFERENCES

- The Complete Reference: Information Security, Mark Rhodes-Ousley, McGrawHill, Second Edition.

*****

UNIT - III

5

SECURE NETWORK DESIGN, NETWORK DEVICE SECURITY

Unit Structure

5.0 OBJECTIVES

- Understand Secure Network Design
- Learn Performance, Availability and Security
- Understand Network Device Security

5.1 INTRODUCTION TO SECURE NETWORK DESIGN

All information systems create risks to an organization, and whether the level of risk introduced is acceptable is ultimately a business decision. Controls such as firewalls, resource isolation, hardened system

configurations, authentication, access control systems, and encryption can be used to help mitigate identified risks to acceptable levels.

5.1.1 Acceptable Risk:

What constitutes an acceptable level of risk depends on the individual organization and its ability to tolerate risk. A risk-averse organization will ultimately accept lower levels of risk and require more security controls in deployed systems. Management's risk tolerance is expressed through the policies, procedures, and guidelines issued to the staff. A complete set of policies outlining management's preferences and its tolerance of information security risks enables employees to make appropriate infrastructure decisions when designing and securing new systems and networks. Thus, the design and configuration of the infrastructure become the enforcement of those documents.

Many enterprises inadvertently violate certain laws without even knowing that they are doing so (for example, storing credit card numbers without taking into account Payment Card Industry Data Security Standard [PCI DSS], or storing patient data without factoring in Health Insurance Portability and Accountability Act [HIPAA] provisions). This modifies the level of residual risk produced after the controls are applied, since the planned controls may not address risks that are not clearly defined prior to control plan development.

5.1.2 Designing Security into a Network:

Security is often an overlooked aspect of network design and attempts at retrofitting security on top of an existing network can be expensive and difficult to implement properly. Separating assets of the differing trust and security requirements should be an integral goal during the design phase of any new project. Aggregating assets that have similar security requirements in dedicated zones allows an organization to use small numbers of network security devices, such as firewalls and intrusion-detection systems to secure and monitor multiple application systems.

Other influences on network design include budgets, availability requirements, network size and scope, future growth expectations, capacity requirements, and management's tolerance of risks. For example, dedicated WAN links to remote offices can be more reliable than virtual private networks (VPNs), but they are costly especially when covering large distances. Fully redundant networks can easily recover from failures, but having duplicate hardware increases costs; and when more routing paths are available, harder it is to secure and segregate traffic flows.

A significant but often missed or under-considered factor in determining an appropriate security design strategy is to identify how the network will be used and what is expected from the business it supports. This design diligence can help avoid expensive and difficult retrofits after the network is implemented. Let's consider some key network design strategies.

Network Design Models:

To understand how the overall design impacts security, let's examine the designs of a shopping mall and an airport. In a shopping mall to make ingress and egress as convenient as possible numerous entrances and exits are provided. However, the large number of entrances and exits makes attempts to control access to the shopping mall expensive and difficult. Screening mechanisms would be required at each door to identify and block unwanted visitors. Furthermore, implementing a screening mechanism isn't the only hurdle; after it is deployed, each mechanism must be kept properly configured and updated to ensure that an unauthorized person doesn't slip through.

In contrast, an airport is designed to funnel all passengers through a small number of well-controlled checkpoints for inspection. Networks built on the shopping mall model are inherently harder to secure than networks designed around the airport model.

The design of an airport does much more than just facilitate the passenger screening performed just inside a terminal. Overall, the airport has a highly compartmentalized design that requires an individual to pass through a security check whenever passing between compartments. Not all screening is explicit—some monitoring is passive, involving cameras and undercover police officers stationed throughout the airport. There are explicit checkpoints between the main terminal and the gate areas as well as between the gate area and the plane. There are security checks for internal airport movements as well and staffs need special access keys to move into the internal areas, such as baggage processing and the tarmac.

An average big-city airport also maintains multiple terminals to handle the traffic load which reduces the impact of a security breach in a single terminal. These smaller, higher-security terminals can have more stringent security checks, and it allows passengers with different security requirements, such as politicians and federal prisoners to be segregated, lowering the risk that one group could affect the other. All of these elements can be translated into network design such as using firewalls and authentication systems for controlling traffic movement around the network, using the network to segregate traffic of differing sensitivity levels, and using monitoring systems to detect unauthorized activities.

5.1.3 Designing an Appropriate Network:

There are invariably numerous requirements and expectations placed upon a network, such as meeting and exceeding the organization's availability and performance requirements, providing a platform that is conducive to securing sensitive network assets, and enabling effective and secure links to other networks. On top of that, the overall network design must provide the ability to grow and support future network requirements. As illustrated earlier with the airport and mall analogies, the overall design of the network will affect an organization's ability to provide levels of security commensurate with any risks associated with the resources or on that network.

To design and maintain a network, network architects and engineers must have a solid understanding of the needs of its users. The best way to do this is to involve those architects and engineers in the application development process. By getting involved early in the development cycle, engineers can suggest more secure designs and topologies and additionally can assure the project team that they have a clear understanding of the security considerations and capabilities. In addition, they can ensure that new projects are more compatible with the existing corporate infrastructure.

Common steps for obtaining such information include meeting with project stakeholders, application and system owners, developers, management, and users. It is important to understand their expectations and needs about performance, security, availability, budget, and the overall importance of the new project. Adequately understanding these elements will ensure that those project goals are met, and appropriate network performance and security controls are included in the design. One of the most common problems encountered in a network implementation is unmet expectations resulting from a difference in assumptions. That's why expectations should be broken down into mutually observable (and measurable) facts as much as possible, so the security designers ensure that there is an explicit agreement with any functional proposals clearly understood and agreed.

5.1.4 The Cost of Security:

Security control mechanisms have expenses associated with their purchase, deployment, and maintenance, and redundantly implementing these systems can increase costs significantly. When deciding on appropriate redundancy and security controls for a given system or network, it is helpful to create several negative scenarios in which a security breach or an outage occurs, to determine the corporation's costs for each occurrence. This risk-model approach should help management determine the value to the corporation of the various security control mechanisms.

For example, what costs are incurred to recover from a security breach or when responding to a system outage outside of normal business hours? Be sure to include cost estimates for direct items such as lost sales, reduced productivity, and replacement costs as well as for indirect items such as damage to the organization's reputation and brand name and the resultant loss of customer confidence. Armed with an approximation of expected loss, corporations can determine appropriate expenditure levels. For example, spending $200,000 to upgrade a trading system to achieve 99.999 percent availability may seem overly expensive on the surface, but it is a trivial expense if system downtime can cost the corporation $250,000 per hour of outage.

5.2 PERFORMANCE

The network will play a huge role in meeting the performance requirements of an organization. Networks are getting faster and faster, evolving from 10 megabits to 100 megabits to gigabit speeds, with 10GE commonly deployed and 40GE, 100GE, and InfiniBand technologies available today. When determining the appropriate network technology, be sure that it can meet the bandwidth requirements projected for three to five years in the future. Otherwise, expensive replacements or upgrades may be required.

Applications and networks that have a low tolerance for latency such as those supporting video and voice streaming will require higher-performance network connections and hardware. What about applications that move data in large chunks (for example, storage snapshots or disk-to-disk offsite replication)? Instead of an expensive, dedicated, high bandwidth connection, it may be more economical to implement links that are burstable, meaning that the provider will allow short bursts of traffic above the normal subscribed rate. If applications will share common network infrastructure components, the design team may also consider implementing Quality of Service (QoS) technologies to prevent one application from consuming too much bandwidth, or to ensure that higher-priority applications always have sufficient bandwidth available.

The legacy Cisco Hierarchical Internetworking model is a common design implemented in large-scale networks today, although many new types of purposed designs have been developed that support emerging technologies like class fabrics, lossless Ethernet, layer two bridging with a trill or IEEE 802.1aq, and other data center–centric technologies.

The three-tier hierarchy still applies to campus networks, but no longer to data centers. This is a “legacy” model socialized by Cisco, but even Cisco has newer thinking for data centers. Networks are becoming much more specialized, and the security thinking for different types of networks is significantly different. The Cisco three-tier model is derived from the Public Switched Telephone Network (PSTN) model, which is in use for much of the world’s telephone infrastructure. The Cisco Hierarchical Internetworking model, depicted in Figure 5.1, uses three main layers commonly referred to as the core, distribution, and access layers:

Core layer:

It forms the network backbone and is focused on moving data as fast as possible between distribution layers. As performance is the core layer’s primary focus, it should not be used to perform CPU-intensive operations such as filtering, compressing, encrypting, or translating network addresses for traffic.

Distribution layer:

It sits between the core and the access layer. This layer is used to aggregate access-layer traffic for transmission into and out of the core.

Access layer:

It is composed of user networking connections.

Filtering, compressing, encrypting, and address-translating operations should be performed at the access and distribution layers.

The Cisco model is highly scalable. As the network grows, additional distribution and access layers can be added seamlessly. As the need for faster connections and more bandwidth arises, the core and distribution equipment can be upgraded as required. This model also assists corporations in achieving higher levels of availability by allowing for the implementation of redundant hardware at the distribution and core layers. And because the network is highly segmented, a single network failure at the access or distribution layers does not affect the entire network.

Although the Cisco three-tier model is perhaps the most commonly known and referenced model for designing LAN environments, it has its limitations and is rapidly being supplanted by newer models aimed at addressing the specific needs of highly virtualized data centers, the specific needs of different industry verticals and the specific needs of cloud computing and multitenancy environments.

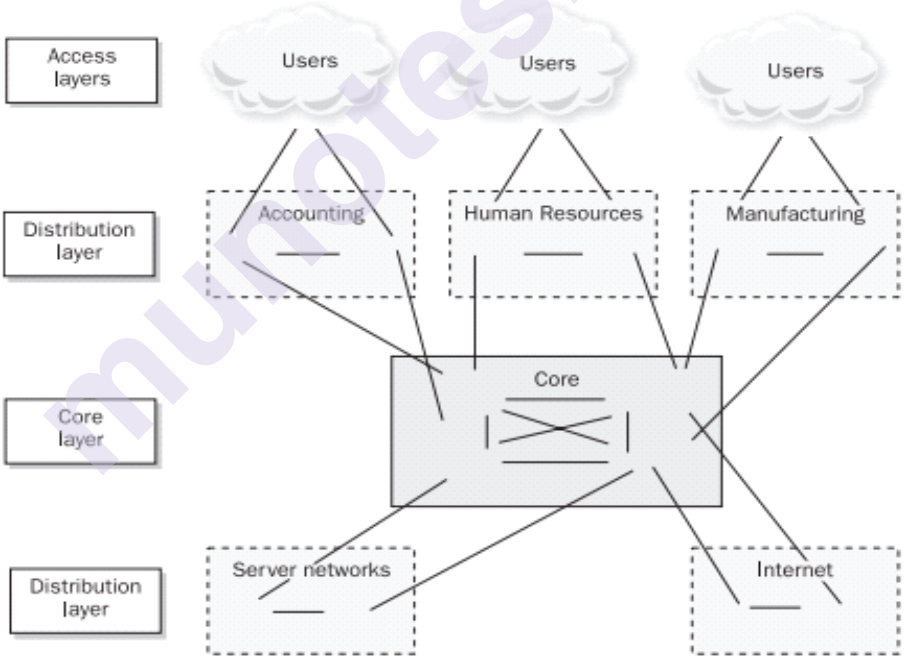


Figure 5.1 The Cisco Hierarchical Internetworking model

5.3 AVAILABILITY

Network availability requires that systems are appropriately resilient and available to users on a timely basis (whenever users require them). The opposite of availability is a denial of service, which is when users cannot access the resources they need on a timely basis. Denial of service can be intentional (for example, the act of malicious individuals) or accidental (such as when hardware or software fails). Business availability needs

have driven some organizations to construct duplicate data centers that perform real-time mirroring of systems and data to provide failover and reduce the risk of a natural disaster or terrorist attack destroying their only data center.

Depending on the specific business and risk factors, redundancy often increases both cost and complexity. Determining the right level of availability and redundancy is an important design element, which is best influenced by a balance between business requirements and resource availability.

The best practice for ensuring availability is to avoid single points of failure within the architecture. This can require redundant and/or failover capabilities at the hardware, network, and application functions. A fully redundant solution can be extremely expensive to deploy and maintain, because as the number of failover mechanisms increases, system complexity increases, which can raise support costs and complicate troubleshooting. Numerous security appliance vendors have failover mechanisms that enable a secondary firewall to take over responsibilities when the primary firewall fails. Beyond firewalls, routers can also be deployed in a high-availability configuration.

Implementing a redundant firewall or router solution is only one step in achieving full high-availability network architecture. For example, a high-availability firewall solution provides no value when both firewalls are plugged into the same switch. The switch becomes a single point of failure and any interruption in its normal operation would take both firewalls off the network, negating any benefit of the firewall failover mechanism. The same holds for a router—if there is only a single router between the firewalls and the rest of the network, the failure of that router would also cause an outage.

A true high-availability design will incorporate redundant hardware components at the switch, network, firewall, and application levels. When eliminating failure points, be sure to consider all possible components. You may want to guarantee reliable power via a battery backup, commonly called an uninterruptible power supply (UPS), or even an emergency generator for potential long-term interruptions. Designers should consider maintaining multiple Internet links to different Internet service providers to insulate an organization from problems at any one provider.

5.4 SECURITY

Each element on a network performs different functions and contains data of differing security requirements. Some devices contain highly sensitive information that could damage an organization if disseminated to unauthorized individuals such as payroll records, internal memorandums, customer lists, and even internal job-costing documents. Other devices have more exposure due to their location on the network. For example,

internal file servers will be protected differently than publicly available web servers.

When designing and implementing security into network and system architectures, it is necessary to identify critical security controls and understand the consequences of a failure in those controls. For example, firewalls protect hosts by limiting what services users can connect to on a given system. Firewalls can allow different sets of users' selective access to different services, such as allowing system administrators to access administrative services while preventing non-administrative users from accessing those same services. This provides an additional level of control over that provided by the administrative mechanisms themselves. By denying a non-administrative user the ability to connect to the administrative service, that user is prevented from mounting an attack directly on that service without first circumventing the firewall.

However, simply restricting users to specific services may be insufficient to achieve the desired level of security. For example, it is necessary to allow traffic through the firewall to connect to various authorized services. For an organization to send and receive an e-mail, firewalls must be configured to permit e-mail traffic. Firewalls have limited capability in preventing attacks directed at authorized applications, so overall network security is dependent on the proper and secure operation of those applications.

Flaws, such as buffer overflows, can allow an attacker to turn a vulnerable server into a conduit through the firewall. Once through the firewall, the attacker can mount attacks against the infrastructure behind the protection of the firewall. If the server is on the internal network, the entire network could be attacked without the protection provided by the firewall, but if the server is on a separate firewalled segment instead of the internal network, only the hosts on the same subnet could be directly attacked. Because all traffic exiting that subnet still must pass back through the firewall, it can still be relied upon to protect any additional communications from this compromised subnet to any other internal subnets.

In addition to the best practice of segmenting the traffic, using the advanced inspection capabilities and application-layer gateways of current-generation firewalls can help protect segmented networks by ensuring that traffic being sent as a particular service over a particular port is well-formed traffic for that service. For example, if a server in a segregated network zone is compromised via an HTTP exploit and the attacker attempts to create a connection to another host within a different firewall zone using SSH but over port 80, the firewall should be able to detect that SSH is not HTTP traffic, and warn or block accordingly (based on how it is configured to behave).

Thus, the network design can increase security by segregating servers from each other with firewalls. However, this is not the only control mechanism that should be used. While it may not be initially obvious, the

proper operation of the service itself is a security control, and limiting the privileges and capabilities of that service provides an additional layer of control. For example, it is good practice to run services without administrative privileges wherever possible.

5.5 NETWORK DEVICE SECURITY

This chapter is about how to use routers and switches to increase the security of the network. The first half of the chapter is about basics of routers and switches, while the second half provides configuration steps for protecting the devices themselves against attacks. Traditionally, routers and switches have been managed by using a command-line interface (CLI), but interfaces have evolved over time toward graphical configuration solutions. CLIs are still available, but web user interfaces (web UIs) have become ubiquitous and are the most used configuration tools these days.

5.5.1 Switch and Router Basics:

The dominant internetworking protocol in use today is known as Transmission Control Protocol/Internet Protocol version 4 (TCP/IP or IPv4), although IPv6 is on the horizon and is deployed in some carrier networks today. TCP/IP provides all the necessary components and mechanisms to transmit data between two computers over a network. TCP/IP is actually a suite of protocols and applications that have discrete functions that map to the Open Systems Interconnection (OSI) model.

5.5.2 MAC Address, IP Address, and ARP:

Each device on a network has two network-related addresses: a layer two address known as the Media Access Control (MAC) address (also known as the hardware address or physical address), and a layer three address known as the IP address. MAC addresses are 48-bit hexadecimal numbers that are uniquely assigned to each hardware network interface by the manufacturer. Each hardware manufacturer has been assigned a range of MAC addresses to use, and each MAC address that has ever been assigned to a physical network interface card (NIC) is globally unique because it allows the underlying communication protocols to select the right system for network communications (although virtual MAC addresses may be used in more than one place because although the algorithms used to generate them are similar and can start with the same reference values, as long as the same two MACs do not appear on the same network segment, they will work).

IPv4 addresses are 32-bit numbers assigned by your network administrator that allow for the creation of logical and ordered addressing on a local network. IPv6 addresses are 128-bit, but like IPv4, each IP address must be unique on a given network. To send traffic, a device must have the destination device's IP address as well as a MAC address. Knowing the destination device's hostname, the sending device can obtain the destination device's IP address using protocols such as Domain Name

Service (DNS). To ascertain a MAC address, the host uses the Address Resolution Protocol (ARP), which functions by sending a broadcast message to the network that basically says, "Who has 192.168.2.10, tell 192.168.2.15." If a host receives that broadcast and knows the answer, it responds with the MAC address: "ARP 192.168.2.10 is at ab:cd:ef:00:01:02." Does this sound like an overly trusting protocol? It was designed by people who had no reason to think anybody would ever abuse it. However, note that no authentication or verification is done for any ARP replies that are received. This facilitates an attack known as ARP poisoning. ARP poisoning is one of the most effective and hard-to-defend attack techniques still in widespread use today.

5.5.3 TCP/IP:

The fundamental purpose of TCP/IP is to provide computers with a method of transmitting data from one computer to another over a network. The purpose of a firewall is to control the passage of TCP/IP packets between hosts and networks.

TCP/IP is a suite of protocols and applications that perform discrete functions corresponding to specific layers of the Open Systems Interconnection (OSI) model. Data transmission using TCP/IP is accomplished by independently transmitting blocks of data across a network in the form of packets, and each layer of the TCP/IP model adds a header to the packet. Depending on the firewall technology in use, the firewall will use the information contained in these headers to make access control decisions. If the firewall is application-aware, as application gateways are, access control decisions can also be made on the data portion or payload of the packet.

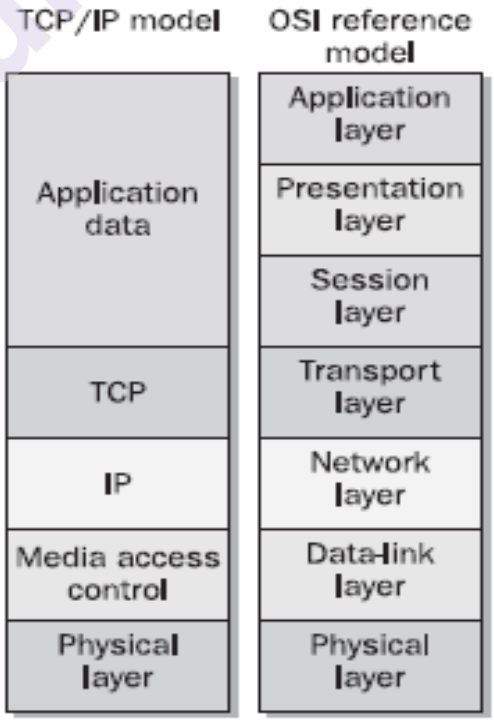


Figure 5.2 The TCP/IP model and the OSI reference model

5.5.4 Brief Overview of the OSI Layer:

The OSI model uses a seven-layer structure to represent the transmission of data from an application residing on one computer to an application residing on another computer. TCP/IP does not strictly follow the seven-layer OSI model, having integrated the upper OSI layers into a single application layer. Figure 5.2 shows a graphical representation of the OSI reference model and its relationship to the TCP/IP implementation.

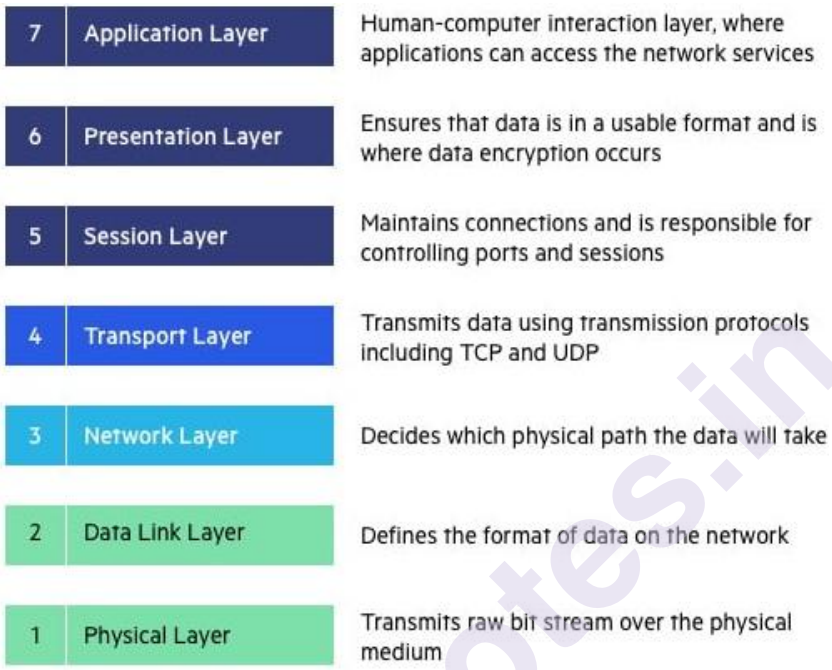


Figure 5.3 OSI Model

1. Physical Layer:

The physical layer is responsible for the physical cable or wireless connection between network nodes. It defines the connector, the electrical cable, or wireless technology connecting the devices, and is responsible for the transmission of the raw data, which is simply a series of 0s and 1s while taking care of bit rate control.

2. Data Link Layer:

The data link layer establishes and terminates a connection between two physically-connected nodes on a network. It breaks up packets into frames and sends them from source to destination. This layer is composed of two parts—Logical Link Control (LLC), which identifies network protocols, performs error checking, and synchronizes frames, and Media Access Control (MAC) which uses MAC addresses to connect devices and define permissions to transmit and receive data.

3. Network Layer:

The network layer has two main functions. One is breaking up segments into network packets and reassembling the packets on the receiving end.

The other is routing packets by discovering the best path across a physical network. The network layer uses network addresses (typically Internet Protocol addresses) to route packets to a destination node.

4. Transport Layer:

The transport layer takes data transferred in the session layer and breaks it into "segments" on the transmitting end. It is responsible for reassembling the segments on the receiving end, and turning them back into data that can be used by the session layer. The transport layer carries out flow control, sending data at a rate that matches the connection speed of the receiving device, and error control, checking if data was received incorrectly and if not, request it again.

5. Session Layer:

The session layer creates communication channels called sessions between devices. It is responsible for opening sessions and ensuring they remain open and functional while data is being transferred and closing them when communication ends. The session layer can also set checkpoints during a data transfer—if the session is interrupted, devices can resume data transfer from the last checkpoint.

6. Presentation Layer:

The presentation layer prepares data for the application layer. It defines how two devices should encode, encrypt and compress data so it is received correctly on the other end. The presentation layer takes any data transmitted by the application layer and prepares it for transmission over the session layer.

7. Application Layer:

The application layer is used by end-user software such as web browsers and email clients. It provides protocols that allow the software to send and receive information and present meaningful data to users. A few examples of application layer protocols are the Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP), and Domain Name System (DNS).

5.5.5 Hubs:

Hubs were dumb devices used to solve the most basic connectivity issue: how to connect more than two devices together. They transmitted packets between devices connected to them, and they functioned by retransmitting each and every packet received on one port out through all of its other ports without storing or remembering any information about the hosts connected to them. This created scalability problems for legacy half-duplex Ethernet networks, because as the number of connected devices and volume of network communications increased, collisions became more frequent, degrading performance.

A collision occurs when two devices transmit a packet onto the network at almost the exact same moment, causing them to overlap and thus mangling them. When this happens, each device must detect the collision and then retransmit its packet in its entirety. As more devices are attached to the same hub, and more hubs are interconnected, the chance that two nodes transmit at the same time increases, and collisions became more frequent. In addition, as the size of the network increases, the distance and time a packet is in transit over the network also increase making collisions even more likely. Thus, it is necessary to keep the size of such networks very small to achieve acceptable levels of performance.

5.5.6 Switches:

Switches are the evolved descendants of the network hub. A network switch connects devices within a network (often a local area network, or LAN) and forwards data packets to and from those devices. Switches were developed to overcome the historical performance shortcomings of hubs. Switches are more intelligent devices that learn the various MAC addresses of connected devices and transmit packets only to the devices they are specifically addressed to. Since each packet is not rebroadcast to every connected device, the likelihood that two packets will collide is significantly reduced. In addition, switches provide a security benefit by reducing the ability to monitor or "sniff" another workstation's traffic. With a hub, every workstation would see all traffic on that hub; with a switch, every workstation sees only its own traffic.

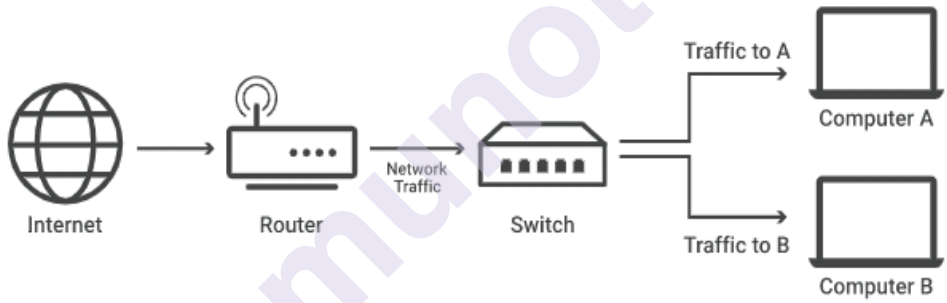


Figure 5.4 Switch

A local area network (LAN) is a group of connected devices within close physical proximity. Home WiFi networks are one common example of a LAN.

When the source wants to send the data packet to the destination, the packet first enters the switch and the switch reads its header and finds the MAC address of the destination to identify the device then it sends the packet out through the appropriate ports that lead to the destination devices. Switch establishes a temporary connection between the source and destination for communication and terminates the connection once the conversation is done. Also, it offers full bandwidth to network traffic going to and from a device at the same time to reduce collision.

5.5.7 Routers:

A router is a device that communicates between the internet and the devices in your home that connects to the internet. As its name implies, it "routes" traffic between the devices and the internet. A router is a key part of your home's internet network. Your laptop, smartphone, smart TV, and other devices can connect to your home Wi-Fi. Routers are primarily used to move traffic between different networks, as well as between different sections of the same network. Routers learn the locations of various networks in two different ways: dynamically via routing protocols and manually via administratively defined static routes. Networks usually use a combination of the two to achieve reliable connectivity between all necessary networks.

Figure 5.5 Router

5.6 NETWORK HARDENING

There are several configuration steps that you can take to ensure the proper operation of your routers and switches. These steps include applying patches as well as taking the time to configure the device for increased security. The more steps and time you take to patch and harden a device, the more secure it will be. You should apply patches and updates released by the product vendor in a timely manner. Quick identification of potential problems and installation of patches to address newly discovered security vulnerabilities can make the difference between a minor inconvenience and a major security incident. To receive timely notification of such vulnerabilities, subscribe to your vendor's e-mail notification services, as well as to general security mailing lists. You will want to keep a special eye out for knowledge base (KB) articles and release notes, which describe changes in device behavior and default settings from one code version to another, in addition to specific vulnerabilities or code bugs being addressed. Ignoring these details can cause potential security issues on your network by negating previous steps you've taken to secure your devices.

5.7 SUMMARY

The ultimate goal of network security is to enable authorized communications while mitigating information risk to acceptable levels. Design elements such as segregating and isolating high-risk or other sensitive assets as well as defining and maintaining a strong network perimeter go a long way toward achieving those goals. As networks become ever more interconnected, a thorough and strongly typed network architecture/design will be required to achieve and maintain a well-secured network. Routers and switches provide a number of mechanisms that, when properly implemented, increase the overall security and performance of the local network.

5.8 QUESTIONS

1. Explain different layers of OSI Model.
2. What is IP Address?
3. Define MAC address.
4. Define switch and routers.

5.9 REFERENCES

- The Complete Reference: Information Security, Mark Rhodes-Ousley, McGrawHill, Second Edition.
- https://www.imperva.com/learn/application-security/osi-model/

*****

6

FIREWALLS

Unit Structure

6.0 OBJECTIVES

- To learn basics of firewall
- Firewall features and functions
- Firewall Configuration
- Tops risks of not having a firewall

6.1 INTRODUCTION

Firewalls have been the most popular and important tools used to secure networks since the early days of interconnected computers. The basic function of a firewall is to screen network traffic to prevent unauthorized access between computer networks.

6.2 OVERVIEW

Firewalls are the first line of defense between the internal network and untrusted networks like the Internet. We should think about firewalls in terms of what you need to protect, so we will achieve the right level of protection for our environment. First introduced conceptually in the late 1980s in a whitepaper from Digital Equipment Corporation, "firewalls" provided a then new and important function to the rapidly growing networks of the day. Before dedicated hardware was commercially available, router-based access control lists were used to provide basic protection and segregation for networks. However, they proved to be inadequate as emerging malware and hacking techniques rapidly developed. Consequently, firewalls evolved over time, so their functionality moved up the OSI stack from layer three to layer seven.

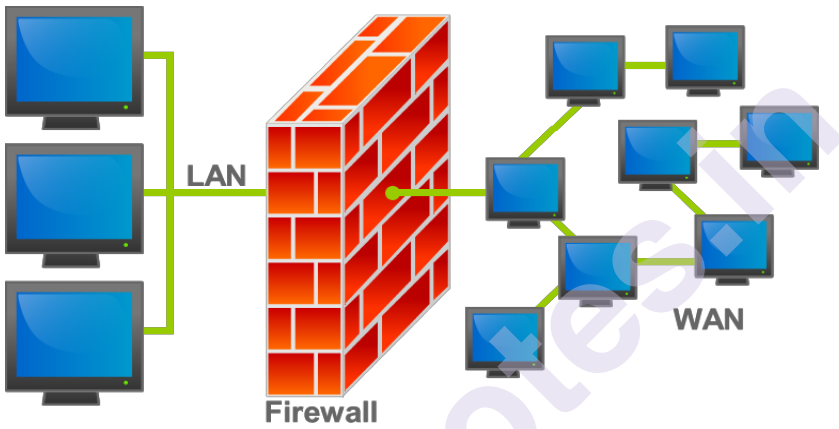


Figure 6.1 Firewall

6.2.1 The Evolution of Firewalls:

First-generation firewalls were simply permitted/deny engines for layer three traffic, working much like a purposed access control list appliance. Originally, first-generation firewalls were primarily used as header-based packet filters, capable of understanding source and destination information up to OSI layer four (ports). However, they could not perform any "intelligent" operations on the traffic other than "allow or deny it from this predefined source IP address to this predefined destination IP address on these predefined TCP and UDP ports."

Second-generation firewalls were able to keep track of active network sessions, putting their functionality effectively at layer four. These were referred to as stateful firewalls or, less commonly, circuit gateways. When an IP address (for example, a desktop computer) is connected to another IP address (say, a web server) on a specific TCP or UDP port, the firewall would enter these identifying characteristics into a table in its memory. This allowed the firewall to keep track of network sessions, which could give it the capability to block man-in-the-middle (MITM) attacks from other IP addresses. In some sophisticated firewalls, a high-availability (HA) pair could swap session tables so that if one firewall failed, a network session could resume through the other firewall.

The third generation of firewalls ventured into the application layer i.e. layer seven. These "application firewalls" were able to decode data inside network traffic streams for certain well-defined, preconfigured applications such as HTTP (the language of the web), DNS (the protocol for IP address lookups), and older, person-to-computer protocols such as FTP and Telnet. Generally, they were unable to decrypt traffic, so they were unable to check protocols like HTTPS and SSH. They were designed with the World Wide Web in mind, which made them well-suited to detect and block website attacks that were generating a great deal of concern at the time, like cross-site scripting and SQL injection.

Consider these in comparison to today's current generation of firewalls (commonly termed the fourth generation), which have the intelligence and capability to look inside packet payloads and understand how applications function. As silicon has increased in speed, advanced router-based firewalls exist today that can provide IP inspection as a software component of a multipurpose router, although they do not provide the speed or sophistication of today's industrial-strength firewalling solutions. In addition, unified threat management (UTM) devices have combined sophisticated, application-layer firewalling capability with antivirus, intrusion detection and prevention, network content filtering, and other security functions. These are true layer seven devices.

Fourth-generation firewalls can run application-layer gateways, which are specifically designed to understand how an application should function and how its traffic should be constructed. There are fifth-generation firewalls, which are internal to hosts and protect the operating system kernel, and some sixth-generation firewalls have been described (meta firewalls), but most network appliances you will find today fall into the generally accepted fourth-generation firewall definition. Some manufacturers call their devices "next-generation firewalls" or "zone-based firewalls," and these essentially function under the same guiding principles of the fourth-generation designs. In this chapter, we primarily focus on fourth-generation firewalls and the key functionality that they enable.

6.2.2 Application Control:

Firewalls have been intended to handle application traffic. Some applications are authorized, and some aren't. For example, web traffic outbound to Internet websites is commonly permitted, while some types of peer-to-peer software are not. On those applications that are allowed, certain behaviors are allowed within the application and others aren't. For instance, web-based meeting and collaboration software might be approved for use on the Internet, but the file-sharing capabilities might be restricted.

First and second-generation firewalls could restrict simple applications that functioned on well-known ports. Back then, applications were well-behaved, communicating on assigned ports that were well-documented, so they were easy to control. But application developers did not always want

to be subject to control, so they devised a simple but effective way to get through the firewall—use port 80. This is known as "tunneling" or "circumventing." Since web traffic uses the HTTP protocol over TCP port 80, it had to be allowed to pass through the firewall unrestricted. There was no practical way to keep track of the millions of IP addresses on the Internet, so applications could freely communicate and their developers were happy.

But then application firewalls came along. These devices could observe the contents of the HTTP traffic traversing port 80 and determine whether it consisted of website-to-browser requests and responses, or something else tunneling through from an application on a local workstation to a remote server. This provided a rudimentary ability to block applications that were prohibited by security policies, but it didn't usually help with controlling application behavior such as allowing voice but not video, or transfer of document files but not photos and movies. Security administrators were concerned about different types of software that could violate security policies, such as:

Peer-to-peer file sharing:

Direct system-to-system communication from an inside workstation to another workstation on the Internet could leak confidential documents, or expose the organization to liability from music and movie copyright violations.

Browser-based file sharing:

Websites that provide Internet file storage via a web browser, allow trusted people inside an organization's network to copy files outside the security administrator's area of control.

Webmail:

Mail services with the ability to add file attachments to messages provide a path to theft and leakage of confidential materials.

Internet proxies and circumvents:

Services running on the Internet or local workstations are explicitly designed to bypass security controls like web filtering.

Remote access:

Remote administration tools are normally used by system administrators to support internal systems from the Internet, which could be abused by Internet attackers.

None of these were easy to control using application-aware firewalls, which could only block broad categories of applications from functioning, or the Internet addresses they needed to connect to, but never with 100 percent effectiveness. That's where fourth-generation firewalls come in. These devices have advanced heuristic application detection and behavior

management capabilities. Circumventing network security controls by using allowed ports isn't effective anymore. Until application developers come up with a new way to circumvent the firewall, the security administrator is back in control.

6.2.3 When Applications Encrypt:

Applications that want to bypass firewalls may encrypt their traffic. This makes the firewall's job more difficult by rendering most of the communication unreadable. Blocking all encrypted traffic isn't feasible except in highly restricted environments where security is more important than application functionality, and a "permit by exception" policy blocks all encrypted application traffic except for that on a whitelist of allowed, known applications.

However, controlling application communications can still be done even if traffic is encrypted, by some of the more advanced fourth-generation firewalls. Applications are easiest to identify by the unique signatures inside their data streams, but there are other identifying features as well. Most have a "handshake protocol" that governs the start of a session, and these usually have an identifiable pattern. Many also have identifiable IP addresses on the Internet they communicate with. Even traffic pattern analysis is possible with advanced heuristic capabilities. A lot of information can be gleaned just from the frequency, size, and timing of communications.

Applications that encrypt their network traffic can be controlled by fourth-generation firewalls, although it's easier to permit or deny the entire application than it is to control the specific functions within it. Today's fourth-generation firewalls have extensive lists of known applications based on extensive research and analysis ready to drag and drop into a policy configuration.

6.3 MUST-HAVE FIREWALL FEATURES

Today's firewalls are expected to do much more than simply block traffic based on the outward appearance of the traffic (such as the TCP or UDP port). As applications have become increasingly complex and adaptive, the firewall has become more sophisticated in an attempt to control those applications. You should expect at least the following capabilities from your firewall.

Application Awareness:

The firewall must be able to process and interpret traffic at least from OSI layers three through seven. At layer three, it should be able to filter by IP address; at layer four by port; at layer five by network sessions; at layer six by data type, and most significantly, at layer seven to properly manage the communications between applications.

Accurate Application Fingerprinting:

The firewall should be able to correctly identify applications, not just based on their outward appearance, but by the internal contents of their network communications as well. Correct application identification is necessary to ensure that all applications are properly covered by the firewall policy configuration.

Granular Application Control:

In addition to allowing or denying communication among applications, the firewall also needs to be able to identify and characterize the features of applications so they can be managed appropriately. File transfer, desktop sharing, voice and video, and in-application games are examples of potentially unwanted features that the firewall should be able to control.

Bandwidth Management (QoS):

The Quality of Service (QoS) of preferred applications, which might include Voice over IP (VoIP) for example, can be managed through the firewall based on real-time network bandwidth availability. If a sporting event is broadcast live via streaming video on a popular website, your firewall should be able to proactively limit or block access so all those people who want to watch it don't bring down your network. The firewall should integrate with other network devices to ensure the highest possible availability for the most critical services.

6.4 CORE FIREWALL FUNCTIONS

Due to their placement within the network infrastructure, firewalls are ideally situated for performing certain functions in addition to controlling application communication. These include Network Address Translation (NAT), which is the process of converting one IP address to another, and traffic logging.

Network Address Translation (NAT):

The primary version of TCP/IP used on the Internet is version 4 (IPv4). Version 4 of TCP/IP was created with an address space of 32 bits divided into four octets, mathematically providing approximately four billion addresses. Strangely enough, this is not sufficient. A newer version of IP, called IPv6, has been developed to overcome this address-space limitation, but it is not yet in widespread deployment.

To conserve IPv4 addresses, RFC 1918 specifies blocks of addresses that will never be used on the Internet. These network ranges are referred to as "private" networks and are identified in Table 6.1. This allows organizations to use these blocks for their corporate networks without worrying about conflicting with an Internet network. However, when these networks are connected to the Internet, they must translate their private IP network addresses into public IP addresses (NAT) to be routable. By

doing this, a large number of hosts behind a firewall can take turns or share a few public addresses when accessing the Internet.

Address	Mask	Range
10.0.0.0	255.0.0.0	10.0.0.0–10.255.255.255
172.16.0.0	255.240.0.0	172.16.0.0–172.31.255.255
192.168.0.0	255.255.0.0	192.168.0.0–192.168.255.255

Table 6.1 Private Addresses Specified in RFC 1918

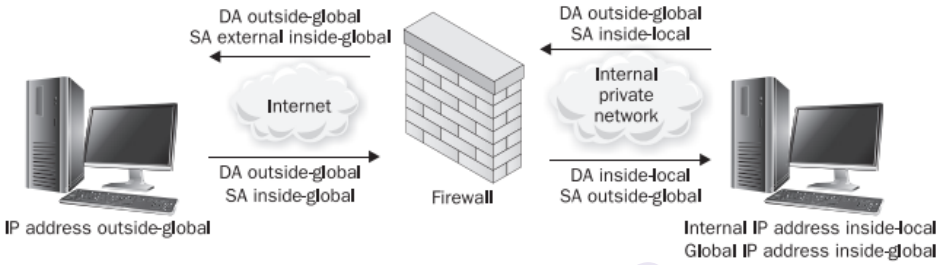


Figure 6.2 Network Address Translation

NAT is usually implemented separately from the policy or rule set in a firewall. It is useful to remember that just because a NAT has been defined to translate addresses between one host and another; it does not mean those hosts will be able to communicate. This is controlled by the policy defined in the firewall rule set.

When hosts have both public and private IP addresses, the IP information contained within a packet header will change depending on where the packet is viewed. For this discussion, the addresses when viewed on the trusted side of the firewall will be referred to as local addresses. Once the packet crosses the firewall and is translated, the addresses will be called the host's global addresses. These terms, as depicted in Figure 6.2, will be used in the following sections to describe the various types and nuances of NAT. In this figure and the other figures in this chapter, the abbreviations "DA" and "SA" refer to "destination address" and "source address" respectively.

Static NAT:

A static NAT configuration always results in the same address translation. The host is defined with one local address and a corresponding global address in a 1:1 relationship, and they don't change. The static NAT translation rewrites the source and destination IP addresses as required for each packet as it travels through the firewall. No other part of the packet is affected. This is typically used for internal servers that need to be reachable from the Internet reliably on an IP address that doesn't change. See Figure 6.3

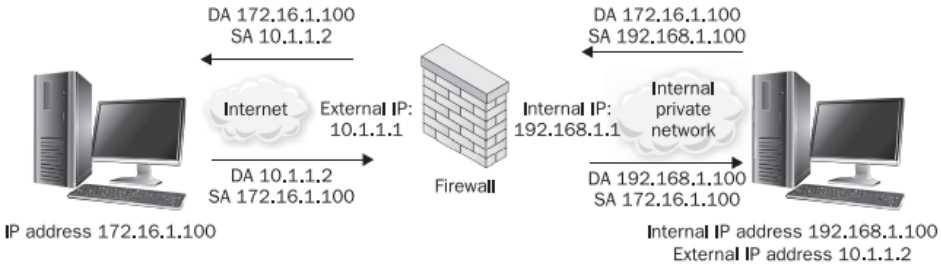


Figure 6.3 NAT replacing global terms with actual IP addresses

Because of this simplistic approach, most protocols will be able to traverse a static NAT without problems. The most common use of static NAT is to provide Internet access to a trusted host inside the firewall perimeter, or inbound access to a specific host, such as a web server that needs to be accessible via a public IP address.

Dynamic NAT:

Dynamic NAT is used to map a group of inside local addresses to one or more global addresses. The global address set is usually smaller than the number of inside local addresses, and the conservation of addresses intended by RFC 1918 is accomplished by overlapping this address space. Dynamic NAT is usually implemented by simply creating static NATs when an inside host sends a packet through the firewall. The NAT is then maintained in the firewall tables until some event causes it to be terminated. This event is often a timer that expires after a predefined amount of inactivity from the inside host, thus removing the NAT entry. This address can then be reused by a different host.

One advantage of dynamic NAT over static NAT is that it provides a constantly changing set of IP addresses from the perspective of an Internet-based attacker, which makes targeting individual systems difficult. The greatest disadvantage of dynamic NAT is the limit on the number of concurrent users on the inside who can access external resources simultaneously. The firewall will simply run out of global addresses and not be able to assign new ones until the idle timers start freeing up global addresses.

Port Address Translation:

With Port Address Translation (PAT), the entire inside local address space can be mapped to a single global address. This is done by modifying the communication port addresses in addition to the source and destination IP addresses. Thus, the firewall can use a single IP address for multiple communications by tracking which ports are associated with which sessions. In the example depicted in Figure 6.4, the sending host initiates a web connection on source port 1045. When the packet traverses the firewall, in addition to replacing the source IP address, the firewall translates the source port to port 5500 and creates an entry in a mapping table for use in translating future packets. When the firewall receives a packet back for destination port 5500, it will know how to translate the

response properly. Using this system, thousands of sessions can be PATed behind a single IP address simultaneously.

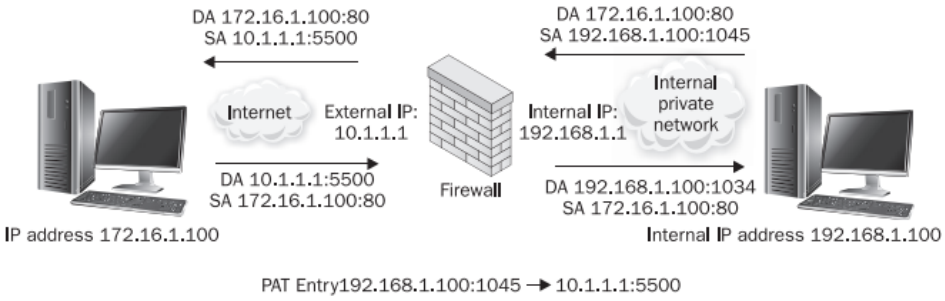


Figure 6.4 an example of Port Address Translation

PAT provides an increased level of security because it cannot be used for incoming connections. However, a downside to PAT is that it limits connection-oriented protocols, such as TCP.

Some firewalls will try to map UDP and ICMP connections, allowing DNS, Network Time Protocol (NTP), and ICMP echo replies to return to the proper host on the inside network. However, even those firewalls that do use PAT on UDP cannot handle all cases. With no defined end of the session, they will usually time out the PAT entry after some predetermined time. This timeout period must be set to be relatively short (from seconds to a few minutes) to avoid filling the PAT table (although, on modern firewalls, the tables used for these sessions, commonly called translation tables, can frequently handle tens of thousands or even millions of sessions).

Connection-oriented protocols have a defined end-of-session built into them that can be picked up by the firewall. The timeout period associated with these protocols can be set to a relatively long period (hours or even days).

Auditing and Logging:

Firewalls are excellent auditors. Given plenty of disk space or remote logging capabilities, they can record any traffic that passes through them. Attack attempts will leave the evidence in logs, and if administrators are watching systems diligently, attacks can be detected before they are successful. Therefore, system activity must be logged and monitored. Firewalls should record system events that are both successful and unsuccessful. Verbose logging and timely reviews of those logs can alert administrators for any suspicious activity before a serious security breach occurs. Since this can generate a huge volume of log traffic, the logs are best sent to a Security Information and Event Management (SIEM) system that can filter, analyze and perform heuristic behavior detection to help the network and security administrators.

6.5 ADDITIONAL FIREWALL CAPABILITIES

Modern firewalls can do more than just manage application communications and behaviors; they can also assist in other areas of network quality and performance. Features vary by manufacturer and brand, but you will probably find that you can solve other problems in your environment with the same firewall you use to secure network traffic.

Application and Website Malware Execution Blocking:

In the old days (just a few years ago), viruses required a user to click on some disguised link or button to execute. If the end users were sophisticated enough to recognize the virus writers' tricks, these viruses wouldn't get very far. Modern malware can execute and spread itself without the intervention of end users. Through automatic, browser-based execution of code (via ActiveX or Java, for example), simply opening a web page can activate a virus. Adobe PDF files can also transmit malware, due to their extensive underlying application framework. Firewalls with advanced anti-malware capabilities should be able to detect these "invisible" malware vectors and stop them in their tracks. They should also be able to block the communication "back home" to a command and control (CnC) server once malware successfully implants itself on a victim system and tries to reach back to its controller for instructions.

Antivirus:

Firewalls that are sophisticated enough to detect malware can (and should) block it on the network. Worms that try to propagate and spread themselves automatically on the network, and malware that tries to "phone home," can be stopped by the firewall, confining their reach. Malware control solutions should be layered, and the firewall can form an important component of a network-based malware-blocking capability to complement your organization's endpoint antivirus software.

Intrusion Detection and Intrusion Prevention:

Firewalls can provide IDS and IPS capabilities at the network perimeter, which can be a useful addition or substitution for standard purpose-built intrusion detection and prevention systems, especially in a layered strategy.

Web Content (URL) Filtering and Caching:

The firewall is optimally positioned on the network to filter access to websites (between an organization's internal networks and the Internet). You can choose to implement a separate URL filtering system or service, or you can get a firewall that has the built-in capability. Today's firewalls are demonstrating web content filtering capabilities that rival those of purpose-built systems, so you may be able to save money by doing the filtering on the firewall—especially if it doesn't cost extra.

E-Mail (Spam) Filtering:

As with web content filtering, modern firewalls can subtract spam from your e-mail messages before they get delivered to your mail server. You can sign up for an external service or buy a purpose-built spam filter instead, but with a firewall that includes this capability, you have another option.

Enhance Network Performance:

Firewalls need to run at "wire speed"—fast enough to avoid bottlenecking application traffic. They should be able to perform all the functions that have been enabled without impacting performance. In addition, firewalls should be able to allocate network bandwidth to the most critical applications to ensure QoS, without sacrificing filtering functionality. As firewall features continue to become more sophisticated, the underlying hardware needs to keep up. If your network has a low tolerance for performance impact, you'll want to consider firewall platforms that are built for speed.

6.6 FIREWALL DESIGN

Firewalls may be software-based or, more commonly, purpose-built appliances. Sometimes the firewalling functions are provided by a collection of several different devices. The specific features of the firewall platform and the design of the network where the firewall lives are key components of securing a network. To be effective, firewalls must be placed in the right locations on the network and configured effectively.

Best practices include

- All communications must pass through the firewall. The effectiveness of the firewall is greatly reduced if an alternative network routing path is available; unauthorized traffic can be sent through a different network path, bypassing the control of the firewall. Think of the firewall in terms of a lock on your front door. It can be the best lock in the world, but if the back door is unlocked, intruders don't have to break the lock on the front door—they can go around it. The door lock is relied upon to prevent unauthorized access through the door, and a firewall is similarly relied upon to prevent access to your network.

- The firewall permits only traffic that is authorized. If the firewall cannot be relied upon to differentiate between authorized and unauthorized traffic, or if it is configured to permit dangerous or unneeded communications, its usefulness is also diminished.

- In a failure or overload situation, a firewall must always fail into a "deny" or closed state, under the principle that it is better to interrupt communications than to leave systems unprotected.

- The firewall must be designed and configured to withstand attacks upon itself. Because the firewall is relied upon to stop attacks, and

nothing else is deployed to protect the firewall against such attacks, it must be hardened and capable of withstanding attacks directly upon itself.

6.7 TYPES OF ATTACK

Before determining exactly what type of firewall you need, you must first understand the nature of security threats that exist. The Internet is one large community, and as in any community, it has both good and bad elements. The bad elements range from incompetent outsiders who do damage unintentionally, to proficient, malicious hackers who mount deliberate assaults on companies using the Internet as their weapon of choice.

Generally, there are three types of attack that could potentially affect your business:

Information theft:

Stealing a company's confidential information such as employee records, customer records, or company's intellectual property.

Information sabotage:

Changing information to damage an individual or company's reputation, such as changing employee's medical or educational records or uploading derogatory content onto your website.

Denial of service (DoS):

Bringing down your company's network or servers so that legitimate users cannot access service and normal company operations such as production are impeded.

6.8 FIREWALL STRENGTHS AND WEAKNESSES

A firewall is just one component of an overall security architecture. Its strengths and weaknesses should be taken into consideration when designing network security.

6.8.1 Firewall Strengths:

Consider the following firewall strengths while designing network security:

- Firewalls are excellent at enforcing security policies. They should be configured to restrict communications to what management has determined and agreed with the business to be acceptable.
- Firewalls are used to restrict access to specific services.
- Firewalls are transparent on the network—no software is needed on end-user workstations.

- Firewalls can provide auditing. Given plenty of disk space or remote logging capabilities, they can log interesting traffic that passes through them.
- Firewalls can alert appropriate people of specified events.

6.8.2 Firewall Weaknesses:

You must also consider the following firewall weaknesses when designing network security:

- Firewalls are only as effective as the rules they are configured to enforce. An overly permissive rule set will diminish the effectiveness of the firewall.
- Firewalls cannot stop social engineering attacks or authorized users intentionally using their access for malicious purposes.
- Firewalls cannot enforce security policies that are absent or undefined.
- Firewalls cannot stop attacks if the traffic does not pass through them.

6.9 FIREWALL PLACEMENT

A firewall is usually located at the network perimeter, directly between the network and any external connections. However, additional firewall systems can be located inside the network perimeter to provide more specific protection to particular hosts with higher security requirements.

6.10 FIREWALL CONFIGURATION

When building a rule set on a firewall, consider the following practices:

- Build rules from most to least specific. Most firewalls process their rule sets from top to bottom and stop processing once a match is made. Putting more specific rules on top prevents a general rule from hiding a specific rule further down the rule set.
- Place the most active rules near the top of the rule set. Screening packets is a processor-intensive operation, and as mentioned earlier, a firewall will stop processing the packet after matching it to a rule. Placing your popular rules first or second, instead of 30th or 31st, will save the processor from going through over 30 rules for every packet. In situations where millions of packets are being processed and rule sets can be thousands of entries in length, CPU savings could be considerable.
- Configure all firewalls to drop “impossible” or “unroutable” packets from the Internet such as those from an outside interface with source addresses matching the internal network, RFC 1918 “private” IP addresses, and broadcast packets. None of these would be expected from the Internet, so if they are seen, they represent unwanted traffic such as that produced by attackers.

6.11 TOP THREE RISKS OF NOT HAVING A FIREWALL

While having a firewall won't ensure that your business will be safe from all manner of attacks, the consequences of not having one are exponentially worse. Look below at the top three risks of not having a firewall:

1. Unlimited Public Access:

Not having a firewall is practically the same as leaving your front door wide open. It's like you're inviting criminals to hack into your network -- and they will. A business without a firewall is easy pickings, as it means everyone can gain access to their network, and they will have no way of monitoring potential threats and untrustworthy traffic.

2. Unrestricted Data Access:

If anyone can waltz into your IT network, they are free to access all your data.

Now, if you think that your small business doesn't have to worry because the data you generate doesn't have value outside your organization, you should seriously reconsider. Your data is valuable and cybercriminals know it.

Without a firewall, you are giving attackers free rein over your information. With that, they can choose to steal your data, leak it to the public, encrypt it and hold it for ransom, or simply delete it. Failing to protect your network with a firewall isn't just a mistake that can cost you a lot of money; it can cost you your business.

3. Network Downtime:

One of the worst possible scenarios you can encounter without a firewall is total network collapse. Without adequate protection, malicious criminals can effectively shut your business down. And that can result in catastrophic damage to your business. Not only can you lose data, but it might also take days or even weeks before your systems can be brought back up and running.

6.12 SUMMARY

This chapter provided an in-depth overview of firewalls, their relevance to applications and OSI layer seven, and their roles in protecting the network. Good security practices dictate that firewalls should be deployed between any two networks of differing security requirements; this includes perimeter connections, as well as connections between sensitive internal networks.

6.13 QUESTIONS

1. Explain the Firewall in detail.
2. Explain Firewall functions.
3. List Firewall Strengths and Weakness.

6.14 REFERENCES

- The Complete Reference: Information Security, Mark Rhodes-Ousley, McGrawHill, Second Edition.

*****

7

WIRELESS NETWORK SECURITY

Unit Structure

7.0 OBJECTIVES

- To learn Radio Frequency Security Basics
- Data-Link Layer Wireless Security Features, Flaws, and Threats
- Wireless Vulnerabilities and Mitigations
- Wireless Network Positioning and Secure Gateways

7.1 INTRODUCTION

Wireless network security is the process of designing, implementing, and ensuring security on a wireless computer network. It is a subset of network security that adds protection for a wireless computer network.

This chapter covers how wireless networking works—because securing a wireless network requires understanding how protocols and signals work—along with wireless threats and countermeasures. We focus on the 802.11 family of wireless LAN protocols collectively known as Wi-Fi,

commonly found in many organizations and households. Wireless security has improved significantly over the past several years, through the use of advanced encryption and access control methods, which means the low-security and simple Wi-Fi targets from ten years ago, are no longer prevalent. Securing a wireless network today can be done through the features of the Wi-Fi products themselves, to the point that today your wireless network will probably be more secure than your wired LAN.

The focus of this chapter is on protecting wireless local area networks.

7.2 RADIO FREQUENCY SECURITY BASICS

In the field of information security, it is an accepted fact that in order to defend against attacks, you have to understand what you're defending. Unfortunately, this fact is not well understood in wireless networking in general because many networks and IT security professionals lack essential knowledge about radio technology. At the same time, radio frequency (RF) experts who switch to the IT field may not be familiar with networking protocols, in particular, complex security-related protocols such as IPSec.

7.2.1 Security Benefits of RF Knowledge:

The following sections describe the security benefits of understanding RF fundamentals.

Proper Network Design:

Security must be taken into account at the earliest stage of network planning and design. This applies to wireless network design even more than to its wired sibling. Poorly designed wireless networks are unfortunately quite common and easy for attackers to spot; they possess low resistance to attacks and tend to slow down to a standstill if network traffic overhead is increased by VPN deployment and rich content such as streaming voice and video.

The Principle of Least Access:

Your wireless LAN (WLAN) should provide coverage where users need it and not anywhere else. The WLAN must be installed and designed in such a way as to encompass your premises' territory and minimize outside signal leakage as much as possible. This ensures that potential attackers have less opportunity to discover your network, less traffic to collect and eavesdrop on, and lower bandwidth to abuse, even if they are successful at circumventing your security measures and managing to associate with the network. It also means the attacker has to stay close to your offices, which makes triangulating and/or physical and video surveillance (CCTV) detection of wireless attackers more likely to succeed.

Distinguishing Security Violations from Malfunctions:

Is it radio interference, or has someone launched a DoS attack? Are these SYN TCP packets coming because the sending host cannot receive SYN-

ACK properly, or is an attacker trying to flood your servers? Why are there so many fragmented packets on the network? Is an attacker running a scanning tool, or is your wireless LAN's maximum transmission unit (MTU) value, which limits the size of network packets, causing frequent retransmits when large packets are sent? The answer is not always obvious. Attacks and malfunctions can appear identical. Most problems on wireless networks can be traced to layer one connectivity issues. Some problems can be caused by neighboring wireless LANs. You shouldn't transmit on the same frequency as your neighbors or one close to it for at least two reasons: interference and the risk of your neighbor accidentally tapping into your data.

Compliance with FCC Regulations:

You don't want to get in trouble with the Federal Communications Commission (FCC) in the United States or its equivalents abroad. Because wireless LAN devices operate in unlicensed bands, these wireless networks can break regulations only by using inappropriately high transmission power. In addition to creating possible legal problems, very high transmission power may send your data further than it needs to go, as discussed in the previous section.

7.2.2 Layer One Security Solutions:

Most issues pertaining to wireless network layer one security can be solved by tuning the transmitter's output power, choosing the right frequency, selecting the correct antennas, and positioning those antennas in the most appropriate way to provide a quality link wherever needed while limiting your network's "fuzzy" borders. Proper implementation of these measures requires knowledge of RF behavior, transmitter power estimation and calculations, and antenna concepts.

Most enterprise controller-based systems with lightweight access points (LWAPs—basically dummies that take all instructions from a central controller) have features like auto frequency switching/hopping, which allows access points to choose the ideal radio frequency depending on current conditions, and dynamic power sensing and adjustment, which raises or lowers the power of the signal so that the communication is optimized without being too weak or too strong. Some systems even have add-on components that can perform real-time frequency management and can use an access point as a "sampler" or air monitor to read the environment around it to provide feedback on how "busy" the air is.

Importance of Antenna Choice and Positioning:

A radio frequency signal is a high-frequency alternating current (AC) passed along the conductor and radiated into the air via an antenna. The emitted waves propagate away from the antenna in a straight line and form RF beams or lobes, which are dependent on antenna horizontal and vertical beam-width values. There are three generic types of antennas, which can be further divided into subtypes:

Omnidirectional	Semidirectional	Highly Directional
Mast mount omni	Patch antenna	Parabolic dish
Pillar mount omni	Panel antenna	Grid antenna
Ground plane omni	Sectorized antenna	
Ceiling mount omni	Yagi antenna	

Antennas are the best friends of wireless network designers, administrators, and consultants. They can also be their worst enemy in the hands of a skillful attacker. They can increase the range of your wireless signal and capture higher volumes of data which the attacker should manage to associate with the target network.

Examples of antenna irradiation patterns are given in Figure 7.1. When choosing necessary antennas, you need to consider antenna irradiation patterns. Get it right, and your coverage is exactly where you need it. Get it wrong, and you'll have dead areas where no one can connect, or you'll exceed the normal boundaries of your environment and broadcast your network beyond reasonable boundaries.

When planning network coverage, remember that irradiation happens in two planes: horizontal and vertical. Try to envision the coverage zone in three dimensions: for example, an omnidirectional beam forms a doughnut-shaped coverage zone with the antenna going vertically through the center of the "doughnut" hole. Sectorized, patch and, panel antennas form a "bubble" typically spreading 60–120 degrees. **Yagi antennas**, named after one of their designers, are directional antennas composed of a dipole and reflector. Yagis form a more narrow "extended bubble" with side and back lobes. Highly directional antennas irradiate a narrowing cone beam, which can reach as far as the visible horizon. Horizontal and vertical planes of semi and highly-directional antennas are often similar in shape but have different beam widths; consult the manufacturer's description of the antenna irradiation pattern before selecting an appropriate antenna for your site.

As you can see from the patterns shown in Figure 7.1, omnidirectional antennas are typically used in point-to-multipoint (hub-and-spoke) wireless network topologies, often together with a variety of semi-directional antennas. Multiple-input multiple-output (MIMO) antennas, which use multiple antenna types to improve coverage, have become common in enterprise systems today.

Yagis are frequently deployed in medium-range point-to-point bridging links, whereas highly directional antennas are used when long-range point-to-point connectivity is required. Highly directional antennas are sometimes used to blast through obstacles such as thick walls. Please note that attackers can also use highly directional dishes to blast through the thick wall of a corporate building, or even through a house that lies in the way of the targeted network. From the top of a hill or a tall building, they can also be used to reach targeted networks 20 to 25 miles away, which makes tracing such attackers hard. On the other hand, at least three highly directional antennas are necessary to triangulate transmitting attackers in order to find their physical position.

Controlling the Range of Your Wireless Devices via Power Output Tuning:

One way to control your wireless signal spread is correct antenna positioning. Another method is to adjust the transmitter power output to suit your networking needs and not the attackers'. Understanding the concept of gain is essential for doing this.

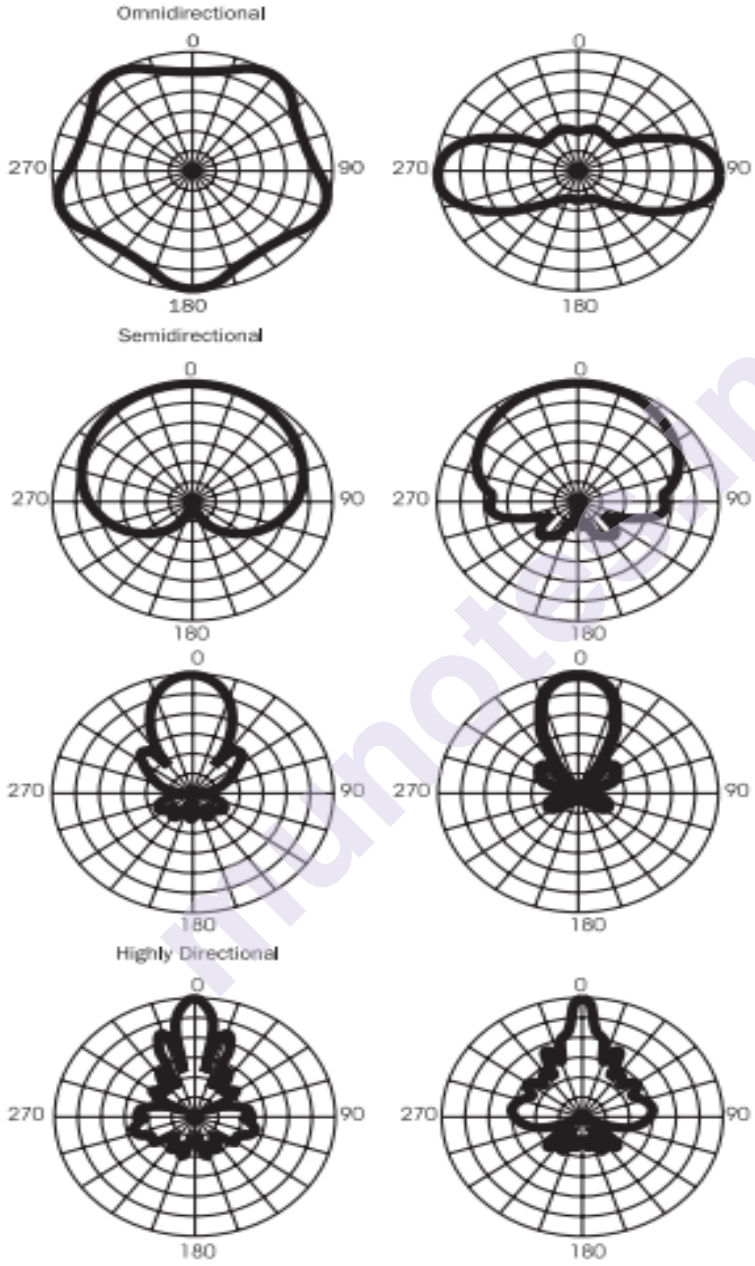


Figure 7.1 Examples of antenna irradiation patterns supplied with quick antenna type–specific beam-width reference values

Gain is a fundamental RF term and has already been referred to several times. Gain describes an increase in RF signal amplitude, as shown in Figure 7.2.

You can achieve a gain in two ways. First, focusing the beam with an antenna increases the signal's amplitude: a narrower beam width means a higher gain. Contrary to popular belief, omnidirectional antennas can possess significant gain reached by decreasing the vertical beam width (squeezing the coverage "doughnut" into a coverage "pancake"). Second, using an amplifier to inject external direct current (DC) power fed into the RF cable (so-called "phantom voltage") can increase gain. Whereas the antenna's direction and position influence where the signal will spread, gain affects how far it will spread by increasing the transmitting power of your wireless devices.

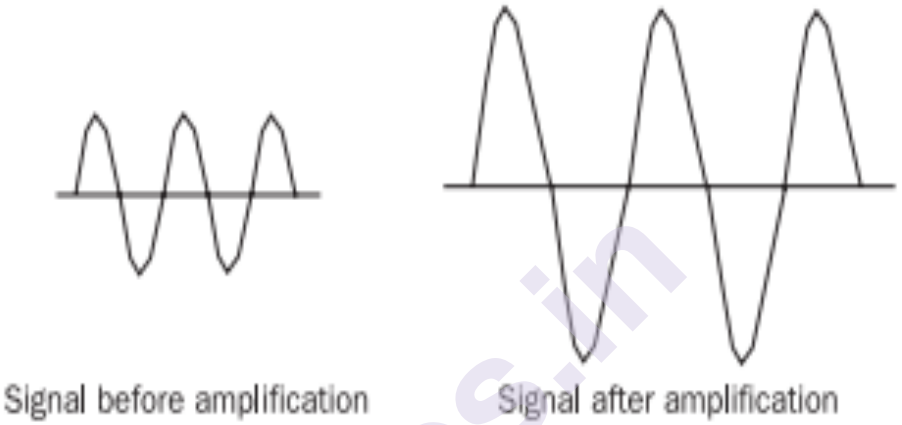


Figure 7.2 Radio frequency signal gain is an increase in the signal's amplitude

The transmitting power output is estimated at two points on a wireless system. The first point is the intentional radiator (IR), which includes the transmitter and all cabling and connectors but excludes the antenna. The second point is the power actually irradiated by the antenna, or equivalent isotropically radiated power (EIRP). Both IR and EIRP output is legally regulated by the U.S. Federal Communications or the European Telecommunications Standards Institute (ETSI). To measure the power of irradiated energy (and the receiving sensitivity of your wireless device), watts (more often milliwatts [mW]) or decibels are used. Power gain and loss (the opposite of gain—a decrease in signal amplitude) are estimated in decibels or, to be more precise, dBm. The m in dBm signifies the reference to 1 mW: 1 mW = 0 dBm. Decibels have a logarithmic relationship with watts: Pdbm = 10 log pmW. Thus, every 3 dB would double or halve the power, and every 10 dB would increase or decrease the power by an order of magnitude. The receiving sensitivity of your wireless devices would be affected in the same way. Antenna gain is estimated in dBi (i stands for isotropic), which is used in the same manner as dBm in RF power calculations.

The best way to find how high your EIRP should be so that it provides a quality link without leaving large areas accessible to attackers is to conduct a site survey with a tool capable of measuring the signal-to-noise ratio (SNR, also estimated in dB as signal strength minus RF noise floor) and pinging remote hosts. Such a tool could be a wireless-enabled laptop

or PDA loaded with the necessary software or a specialized wireless site survey device.

You can estimate EIRP and loss mathematically before running the actual site survey, taking into account the events depicted in Figure 7.3.

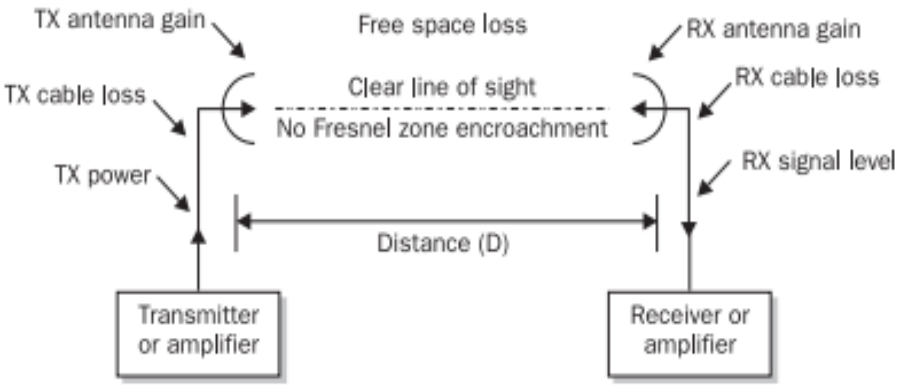


Figure 7.3 Wireless link power gain and loss

Free space path loss is the biggest cause of energy loss on a wireless network. It happens because of the radio wavefront broadening and transmitted signal dispersion (think of a force decreasing when it is applied to a larger surface area). Free space path loss is calculated as 36.56 + 20 $\log_{10}$ (frequency in GHz) +20 $\log_{10}$ (distance in miles). The Fresnel zone in Figure 7.4 refers to a set of specific areas around the line of sight between two wireless hosts. You can try to imagine it as a set of elliptical spheres surrounding a straight line between two wireless transmitters, building a somewhat rugby ball–shaped zone along this line. The Fresnel zone is essential for wireless link integrity since any objects obstructing this zone by more than 20 percent introduce RF interference and can cause signal degradation or even complete loss. At its widest point, the radius of the Fresnel zone can be estimated as

$$43.3 \times (\text{link distance in miles} / (4 \times \text{signal frequency in GHz}))$$

Free space path loss and Fresnel zone calculators are available online at the websites already mentioned when referring to RF power output calculations. In the real world, the power loss between hosts on a wireless network is difficult to predict, owing to the likely objects in the Fresnel zone (for example, trees or office walls) and the interaction of radio waves with these objects and other entities in the whole coverage area. Such interactions can include signal reflection, refraction, and scattering (see Figure 7.4).

Apart from weakening the signal, these interactions can leak out your network traffic to unpredicted areas, making network discovery more likely and giving potential attackers the opportunity to eavesdrop on network traffic where no one expects the traffic's (and the attackers') presence.

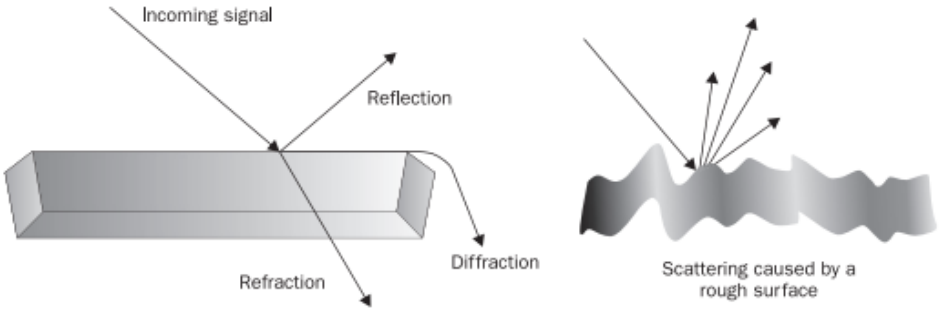


Figure 7.4 Electromagnetic wave-object interactions

Although you may wonder what the relationship is between legal limitations on acceptable wireless power output and wireless security, you don't want to be a major source of interference in your area and end up on the same side of the law as the attackers. Besides attackers are not limited by the FCC—if one is going to break the law anyway, why care about FCC rules and regulations? This point is important when reviewing layer one DoS (jamming) and layer one man-in-the-middle attacks on wireless networks. Although a wireless systems administrator cannot "outpower" attackers by exceeding the legal power limits, he or she can implement other measures, such as a wireless IDS capable of detecting layer one anomalies like sudden RF power surges or signal quality failures on the monitored network, to alleviate the problem.

Interference, Jamming, and the Coexistence of Spread Spectrum Wireless Networks:

The basic concepts of spread spectrum communications are necessary for an understanding of interference, jamming, and the coexistence of wireless networks. Spread spectrum refers to wide-frequency low-power transmission, as opposed to narrowband transmission, which uses just enough spectrum to carry the signal and has a very large SNR (see Figure 7.5).

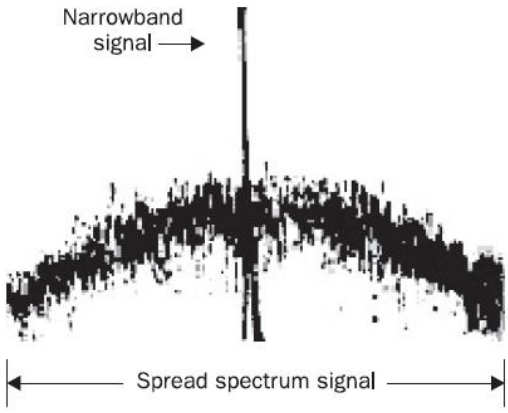


Figure 7.5 Spread spectrum versus narrowband transmission

All 802.11 and 802.15 IEEE standards–defined wireless networks employ spread spectrum band technology. This technology was originally developed during World War II, with security being the primary

development aim. Anyone sweeping across the frequency range with a wideband scanner who doesn't know how the data is carried by the spread spectrum signal and which frequencies are used will perceive such a signal as white noise. Using spread spectrum technology in military communications is a good example of "security through obscurity" that works and is based on very specific equipment compatibility.

In everyday commercial and hobbyist wireless nets, however, this obscurity is not possible. The devices used must be highly compatible, interoperable, and standards-compliant (in fact, interoperability is the main aim of the Wireless Ethernet Compatibility Alliance (WECA) "WiFi" certification for wireless hardware devices, which many confuse with the IEEE 802.11b data-link layer protocol standard). When the link between communicating devices is established, the two devices must agree on a variety of parameters such as communication channels. Such agreement is done via unencrypted frames sent by both parties. Anyone running a wireless sniffer can determine the characteristics of a wireless link after capturing a few management frames off the air. Thus, the only security advantage brought to civil wireless networks by implementing spread spectrum technology is the heightened resistance of these networks to interference and jamming as compared to narrowband transmission.

There are two ways to implement spread spectrum communications:

- Frequency hopping spread spectrum (FHSS)
- Direct sequence spread spectrum (DSSS)

In FHSS, a pseudorandom sequence of frequency changes (hops) is followed by all hosts participating in a wireless network (see Figure 7.6).

The carrier remains at a given frequency for a dwell time period and then hops to another frequency (spending a hop time to do it); the sequence is repeated when the list of frequencies to hop through is exhausted. FHSS was the first spread spectrum implementation technology proposed.

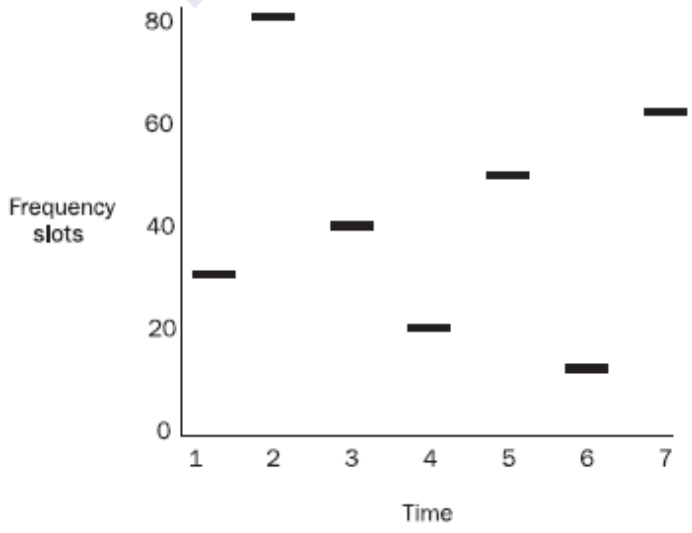


Figure 7.6 FHSS frequency hopping

It is used by legacy 1–2 Mbps 802.11 FHSS networks and most importantly, 802.15 networks (Bluetooth). Bluetooth hops 1600 times per second (~625 μs dwell time) and must hop through at least 75 MHz of bandwidth in the middle ISM band. As such, Bluetooth is very resistant to radio interference unless the interfering signal covers the whole middle ISM band. At the same time, Bluetooth devices (Class 3 transmitters) introduce wideband interference capable of disrupting 802.11, 802.11b, and 802.11g LANs. Thus, a Bluetooth-enabled phone, PDA, or laptop can be an efficient (unintentional or intentional) wideband DoS/jamming tool against other middle ISM band wireless networks.

As for interference issues arising from using multiple Bluetooth networks in the same area, it is theoretically possible to keep 26 Bluetooth networks in the same area owing to the different frequency hopping sequences on these networks. In practice, however, exceeding 15 networks per area is not recommended, but the time when widespread Bluetooth use will create such a density of networks is coming—and is closer than it seems—colleges now plan for 7 devices per user for campus-provided wireless networks. You can imagine that in a dorm room with 4 to 6 tenants in proximity, the number of Bluetooth networks could easily exceed 15 networks.

The 802.11 range of network uses DSSS. As compared to FHSS networks (with a maximum 5 MHz–wide carrier frequency), DSSS networks use wider channels (802.11b/g: 22 MHz, 802.11a: 20 MHz), which allow higher data transmission rates. On the other hand, because the transmission on a DSSS network goes through a single 20 to 22-MHz channel and not the whole ISM/UNII band range or the 75 MHz defined by the FCC for FHSS networks, DSSS networks are more vulnerable to interference and jamming. An 802.11b or g LAN would suffer from colocation with a Bluetooth network to a greater extent than the network would be negatively affected by the 802.11b/g LAN.

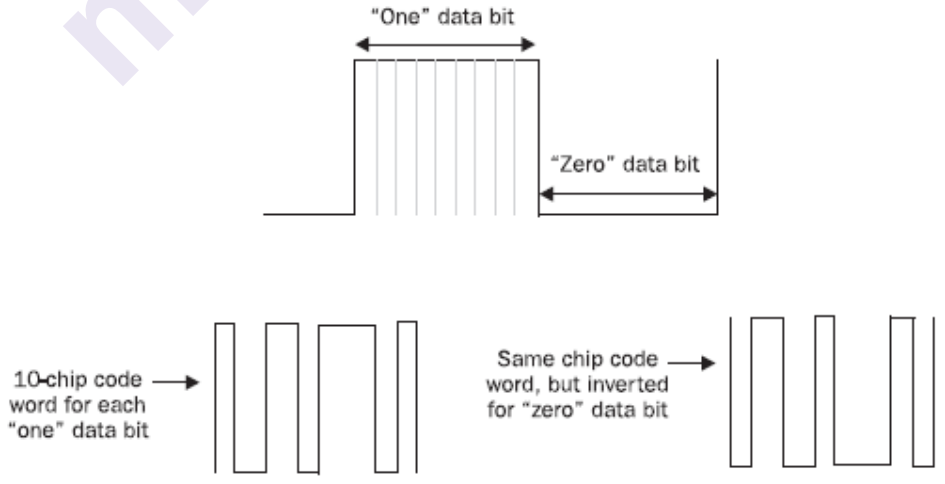
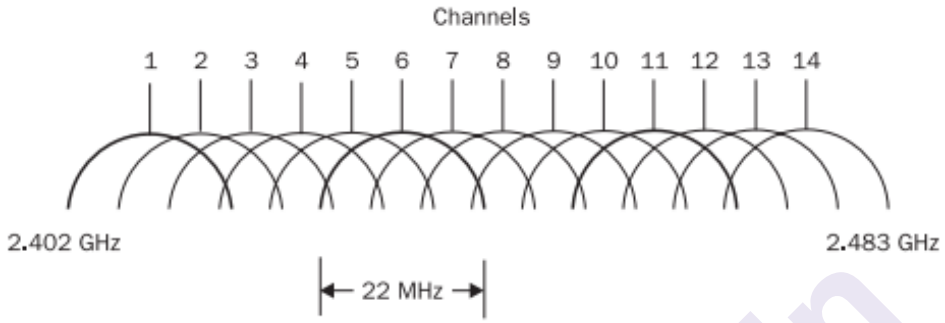


Figure 7.7 DSSS data "hiding" and transmission

UNII band DSSS channels are split by 5 MHz between the channel "margins"; thus, they do not overlap. On the contrary, middle ISM band

DSSS channels are split by the 5-MHz distance between the middle of each channel, which means severe channel overlapping takes place. The 802.11b/g channel width is 22 MHz, so you need at least 5 channels (5×5 MHz = 25 MHz > 22 MHz) between two nonoverlapping channels, or so the theory goes. In reality, even these channels would interfere with each other for a variety of reasons. In the U.S., you can use 11 802.11b/g channels, so the maximum number of coallocated access points is three, taking channels 1, 6, and 11, as the following illustration of the 802.11b/g frequency channels allocation shows.



In Europe, 13 channels are allocated for 802.11b/g use, making access point coallocation more flexible (however, only the channels from 10 to 13 are used in France and 10 to 11 in Spain). All 14 channels can be used in Japan. Channel allocation has high relevance to the much-discussed issue of rogue access points. There are various definitions for a "rogue access point" and, therefore, different ways of dealing with the problem:

Access points and bridges that belong to neighboring LANs and interfere with your LAN by operating on the same or overlapping channels:

Solution: Be a good neighbor and reach an agreement with other users on the channels used so they do not overlap. Ensure your data is encrypted and an authentication mechanism is in place. Advise your neighbors to do the same if their network appears to be insecure.

Note that interference created by access points operating on close channels (such as 6 and 7) is actually higher than interference created by two access points operating on the same channel. Nevertheless, two or more access points operating on the same channel do produce significant signal degradation. Unfortunately, many network administrators who do not understand RF basics tend to think that all access points belonging to the same network or organization must use the same channel, which is not true.

Access points, bridges, USB adapters, and other wireless devices installed by users without permission from enterprise IT management:

Solution: Have a strictly defined ban on unauthorized wireless devices in your corporate security policy and be sure all employees are aware of the policy contents. Detect wireless devices in the area by using wireless

sniffers or specific wireless tools and appliances. Remove discovered unwanted devices and check if the traffic that originated from such devices produced any alerts in logs.

Access points or other wireless devices installed by intruders provide a back channel into the corporate LAN, effectively bypassing egress filtering on the firewall:

Solution: This is a physical security breach and should be treated as such. Apart from finding and removing the device and analyzing logs (as in the preceding point), treat the rogue device as serious evidence. Handle it with care to preserve attackers' fingerprints, place it in a sealed bag, and label the bag with a note showing the time of discovery as well as the credentials of the person who sealed it. Investigate if someone has seen the potential intruder and check the information provided by CCTV.

Outside wireless access points and bridges employed by attackers to launch man-in-the-middle attacks:

This is a "red alert" situation and indicates skill and determination on the part of the attacker. The access point can be installed in the attacker's car and plugged into the car accumulator battery, or the attacker could be using it from a neighboring apartment or hotel room. Alternatively (and more comfortably for an attacker), a PCMCIA card can be set to act as an access point. An attacker going after a public hotspot may try to imitate the hotspot user authentication interface in order to capture the login names and passwords of unsuspecting users.

Solution: Above all, such attacks indicate that the assaulted network was wide open or data encryption and user authentication mechanisms were bypassed. Deploy your wireless network wisely, implementing security safeguards. If the attack still takes place, consider bringing down the wireless network and physically locating the attacker. To achieve the latter aim, contact a specialized wireless security firm capable of attacker triangulation.

7.3 DATA-LINK LAYER WIRELESS SECURITY FEATURES, FLAWS, AND THREATS

The peculiarities of physical layer operations, as well as the expected wireless network topology and size, determined the design of data-link layer protocols and associated security features for wireless communications. Unfortunately, reality rarely meets the designer's expectations. Wireless LANs were initially developed for limited-size networks and short-to-medium point-to-point bridging links.

7.3.1 802.11 and 802.15 Data-Link Layer in a Nutshell:

Here, we'll briefly review layer two operations of commonly used wireless networks such as 802.11 LANs and Bluetooth networks. Despite the common use of the terms "wireless Ethernet" and "ethX" as wireless

interface designations, the data-link layer on 802.11 networks is quite different from Ethernet frames, as Figure 7.8 demonstrates.

A wireless LAN's mode of operation is also dissimilar to that of an Ethernet. As a radio transceiver can only transmit or receive at a given time on a given frequency, all 802.11-compliant networks are half-duplex. Whereas an access point is a translational bridge in relation to the wired network it may be connected to, for wireless network clients, the access point acts as a hub, making packet sniffing an easy task. Because detecting collisions on a wireless network is not possible, the Carrier Sense Media Access/Collision Avoidance (CSMA/CA) algorithm is used on wireless LANs instead of Ethernet's CSMA/ CD algorithm. CSMA/CA is based on receiving a positive ACK for every successfully transmitted frame and retransmitting data if the ACK frame is not received. On wired networks, by plugging in the cable, you are associated with the network. On wireless networks, you can't do this, and the exchange of association request and response frames followed by the exchange of authentication request and response frames is required. Before requesting association, wireless hosts have to discover each other. Such discovery is done by means of passive scanning (listening for beacon frames sent by access points or ad hoc wireless hosts on all channels) or active scanning (sending probe request frames and receiving back probe responses). If a wireless host loses connectivity to the network, another exchange of reassociation, request and response frames takes place. Finally, a deauthentication frame can be sent to an undesirable host.

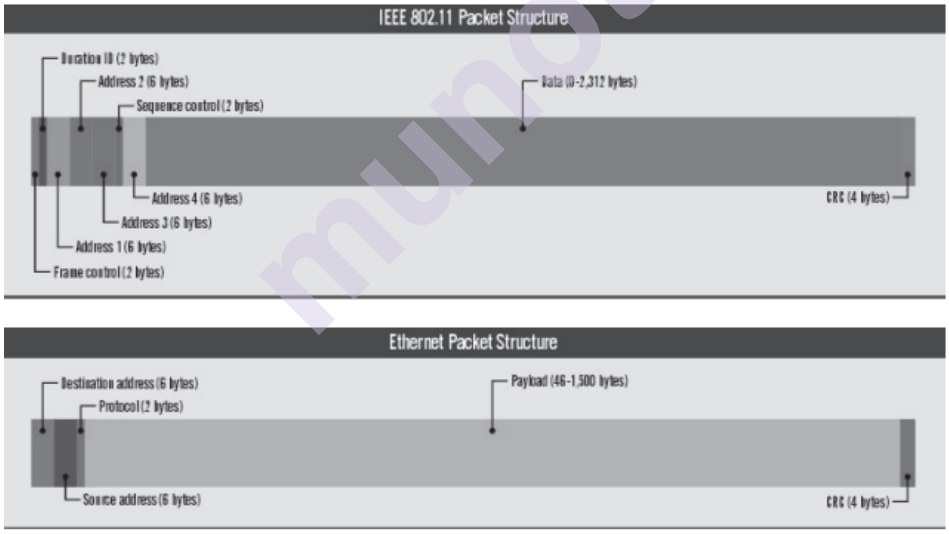


Figure 7.8 Comparison between 802.11 and 802.3 frames

MIMO, in the context of Wi-Fi, is still half-duplex, but MIMO allows a fancy way to "hide" or get around the duplex limitation by simultaneously transmitting in both directions (send and receive) on different antennas.

Bluetooth wireless networks can function in circuit-switching (voice communications) and packet-switching (TCP/IP) modes, which can be used simultaneously. The Bluetooth stack is more complicated than its 802.11 counterparts, spanning all the OSI model layers (see Figure 7.9).

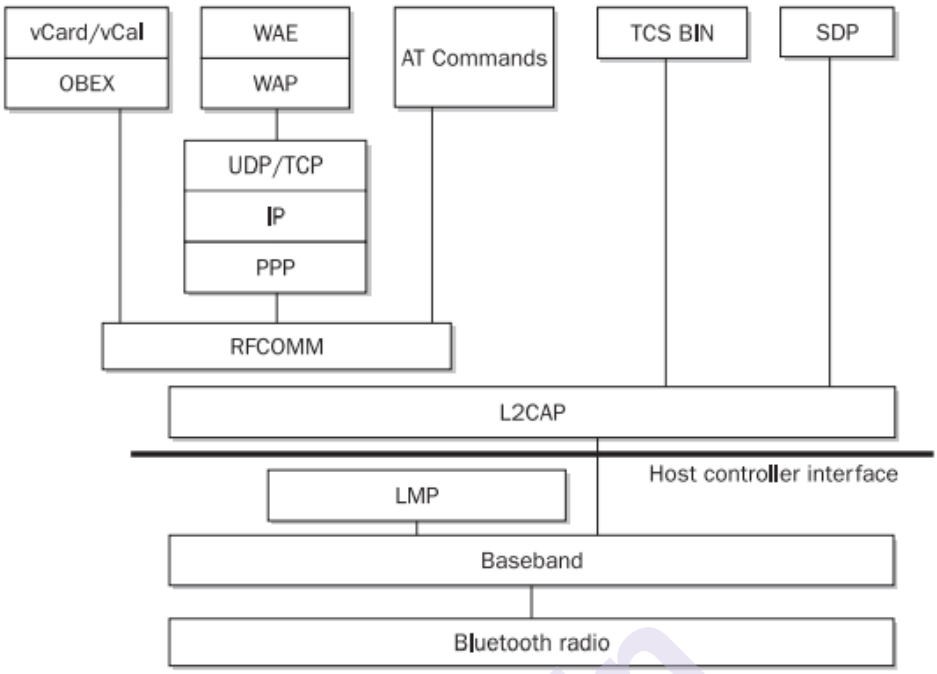


Figure 7.9 Bluetooth protocol stack

The Link Manager Protocol (LMP) is responsible for setting up the link between two Bluetooth devices. It decides and controls the packet size, as well as provides security services such as authentication and encryption using link and encryption keys. The Logical Link Control and Adaptation Protocol (L2CAP) is responsible for controlling the upper layer protocols. RFCOMM is a cable replacement protocol that interfaces with the core Bluetooth protocols. The Service Discovery Protocol (SDP) is present so that Bluetooth-enabled devices can gather information about device types, services, and service specifications to set up the connection between devices. Finally, there are a variety of application-layer protocols such as TCS BINARY and AT Commands; these are telephony control protocols that allow modem and fax services over Bluetooth.

7.3.2 802.11 and 802.15 Data-Link Layer Vulnerabilities and Threats:

The main problem with layer two wireless protocols is that in both 802.11 and 802.15 standards, the management frames are neither encrypted nor authenticated. Anyone can log, analyze and transmit them without necessarily being associated with the target network. While intercepting management frames is not the same as intercepting sensitive data on the network, it can still provide a wealth of information, including network SSIDs (basically, the network name), wireless hosts' MAC addresses, DSSS LAN channels in use, FHCC frequency hop patterns, and so on. Every Bluetooth device has a unique ID transmitted in clear text in the management frames. Thus, eavesdropping on these frames can be helpful in tracking such a device and its user. Preventing this is hard—short of turning off the Bluetooth device entirely.

Unfortunately, the information presented by management frames is only a tiny fraction of the problem. The attacker can easily knock wireless hosts

offline by sending deauthenticate and disassociated frames. Even worse, the attacker can insert his or her machine as a rogue access point by spoofing the real access point's MAC and IP addresses, providing a different channel to associate, and then sending a disassociate frame to the target host(s).

7.3.3 Closed-System SSIDs, MAC Filtering, and Protocol Filtering:

Common nonstandard wireless LAN safeguards include closed-system SSIDs, MAC address filtering, and protocol filtering.

Closed-system SSID is a feature of many higher-end wireless access points and bridges. It refers to the removal of SSID from the beacon frames and/or probe response frames, thus requiring the client hosts to have a correct SSID in order to associate. This turns SSID into a form of shared authentication password. Closed-system SSIDs can be found in management frames other than beacons and probe responses, however. Just as in the case of shared key authentication mode, wireless hosts can be forced to disassociate in order to capture the SSID in the management frame's underlying reassociation process. Attackers can easily circumvent closed-system SSID security by using deassociation/deauthentication frames.

MAC filtering, unlike closed-system SSID, is a common feature that practically every modern access point supports. It does not provide data confidentiality and is easily bypassed (again, an attacker can force the target host to disassociate without waiting for the host to go offline so its MAC address can be assumed). Nevertheless, MAC filtering may stop script kiddie (unsophisticated) attackers from associating with the network.

Finally, protocol filtering is less common than closed systems and MAC address filtering; it is useful only in specific situations and when it is sufficiently selective. For example, when the wireless hosts only need web and mail traffic, you can filter all other protocols and use the built-in encryption capabilities of web and mail servers to provide a sufficient degree of data confidentiality. Alternatively, SSH port forwarding can be used. Protocol filtering combined with secure layer six protocols can provide a good security solution for wireless LANs built for handheld users with low-CPU power devices limited to a specific task (barcode scanning, browsing the corporate website for updates, and so on)

7.3.4 Built-in Bluetooth Network Data-Link Security and Threats:

Bluetooth has a well-thought-out security mechanism covering both data authentication and confidentiality. This mechanism relies on four entities: two 128-bit shared keys (one for encryption and one for authentication), one 128-bit random number generated for every transaction, and one 48-bit IEEE public address (BD_ADDR) unique to each Bluetooth device. Setting up a secure Bluetooth communication channel involves five steps:

1. An initialization key is generated by each device using the random number, BD_ADDR, and shared PIN.
2. Authentication keys (sometimes called link keys) are generated by both ends.
3. The authentication keys are exchanged using the initialization key, which is then discarded.
4. Mutual authentication via a challenge-response scheme takes place.
5. Encryption keys are generated from authentication keys, BD_ADDR, and a 128-bit random number.

Streaming cipher E0 is used to encrypt data on Bluetooth networks. A modification of the SAFER+ cipher is used to generate the authentication keys. Three Bluetooth security modes are known: insecure mode 1, service-level security mode 2, and link-level enforced security mode 3. Mode 3 is the most secure and should be used where possible.

7.4 WIRELESS VULNERABILITIES AND MITIGATIONS

Since Wi-Fi primarily operates at layer two in the OSI stack, most of the attacks against it occur at layer two. But wireless attacks, such as jamming, can also occur at layer one. In this section, we describe five types of wireless attacks.

7.4.1 Wired Side Leakage:

Network attacks—whether on the wired or wireless network—typically begin with some form of reconnaissance. On wireless networks, reconnaissance involves promiscuously listening for wireless packets using a wireless sniffer so the attacker can begin to develop a footprint of the wireless network. We will ideally focus on layer two packets, whereby we are not connected (associated) to an access point. If the attackers were associated with an access point, then he or she could sniff layer three and above.

Broadcast and multicast traffic run rampant on most wired networks, thanks to protocols such as NetBIOS, OSPF, and HSRP, among others that were designed to be chatty about their topology information because they were envisioned to be used only on protected internal networks. What many administrators don't realize is that when they connect wireless networks to their wired networks, this broadcast and multicast traffic can leak into the wireless airspace, as shown in Figure 7-10, if not properly segmented and firewalled. Most access points and wireless switches allow this traffic to leak into the airspace without being blocked. Figure 17-10 illustrates this concept with a network device that is connected to an AP via a wired network, leaking internal protocol communications onto the airwaves. Unfortunately, this traffic may reveal network topology, device types, usernames, and even passwords!

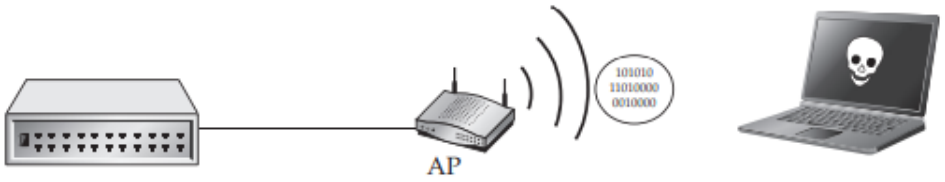


Figure 7.10 Network device traffic can leak onto the wireless airspace

For instance, Cisco's Hot Standby Router Protocol (HSRP), which is used for gateway failover, sends multicast packets. By default, these packets broadcast heartbeat messages back and forth that include the hot standby password for the router in clear text. When these packets leak from the wired network to the wireless airspace, they reveal information about the network topology as well as the password, as shown in Figure 7-11.

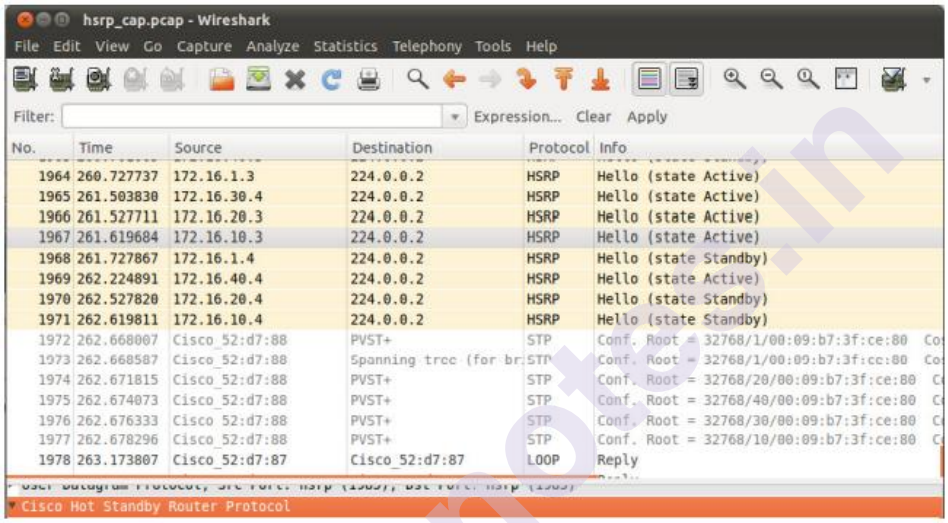


Figure 7.11 A password is revealed by an internal routing protocol via wireless.

When deploying wireless, you need to ensure that, like a firewall, ingress, as well as egress is considered. Outbound traffic on the wireless switch and access point should be properly filtered of broadcast traffic to prevent this sensitive wired traffic from leaking into the local airspace. A wireless intrusion prevention system (IPS) can help to identify this wired-side leakage by monitoring packets for signs of data leakage, so administrators can block any leaks on their access points, wireless switches, or firewalls.

7.4.2 Rogue Access Points:

The most common type of rogue access point involves a user who brings a consumer-grade access point like a Linksys router into the office. Many organizations attempt to detect rogue APs through wireless assessments. It is important to note that although you may detect access points in your vicinity, it is equally important to validate if they are connected to your physical network. The definition of a rogue AP is an unsanctioned wireless access point connected to your physical network. Any other visible AP that's not yours is simply a neighboring access point.

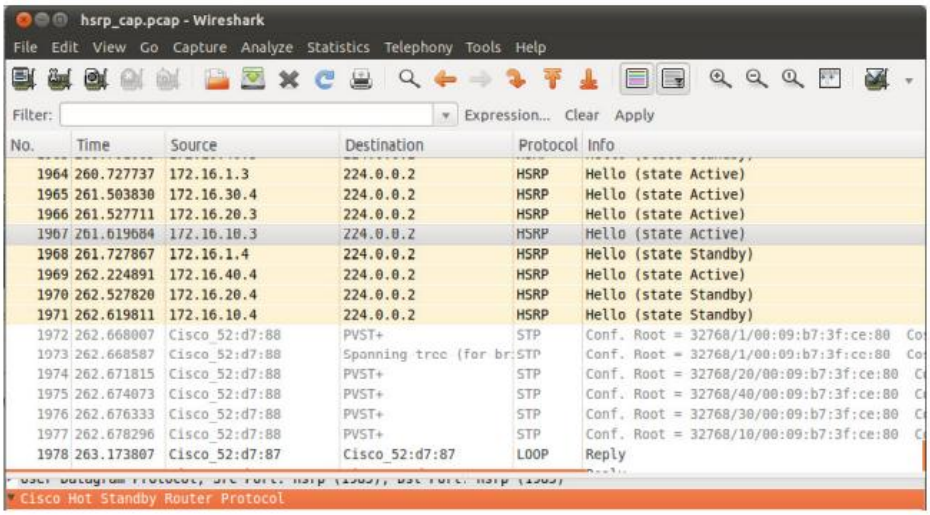


Figure 7.11 A password is revealed by an internal routing protocol via wireless

Vetting out the potential rogue APs requires some prior knowledge of the legitimate wireless environment and sanctioned access points. This approach for detecting rogue APs involves determining the anomalous access points in the environment and, therefore, is really the best-effort approach. As mentioned earlier, this approach doesn't necessarily confirm whether the access points are physically connected to your network. That requires assessing the wired side as well and then correlating the wired assessment to the wireless assessment. Otherwise, the only other option is to check each physical access point to determine if the anomalous AP is connected to your network. Doing this can be impractical for a large assessment. For this reason, wireless IPSs are far more effective at detecting rogue APs. A wireless IPS correlates what it sees with its wireless sensors to what it sees on the wired side. Through a variety of algorithms, it determines if the access point is truly a rogue access point, one that is physically connected to the network.

Even quarterly spot checks for rogue access points still give malicious hackers a huge window of opportunity, leaving days if not months for someone to plug in a rogue access point, perform a compromise, and then remove it without ever being detected.

7.4.3 Misconfigured Access Points:

Enterprise wireless LAN deployments can be riddled with misconfigurations. Human error coupled with different administrators installing the access points and switches can lead to a variety of misconfigurations. For example, an unsaved configuration change can allow a device to return to its factory default setting if, say, the device reboots during a power outage. And numerous other misconfigurations can lead to a plethora of vulnerabilities. Therefore, these devices must be monitored for configurations that are in line with your policies. Some of this monitoring can be done on the wired side with WLAN management products. Additionally, mature wireless IPS products can also monitor for

misconfigured access points if you predefine a policy within the wireless IPS to monitor for devices not compliant with policy.

Modern systems have different considerations—the controller-based approach largely prevents this issue, but some organizations, especially smaller ones, will still face this type of problem. Human error on the controller side poses a larger and more significant risk—all the access points will have a problem or configuration vulnerability, not just one.

7.4.4 Wireless Phishing:

Since organizations are becoming more disciplined with fortifying their wireless networks, trends indicate that wireless users have become the low-hanging fruit. Enforcing secure Wi-Fi usage when it concerns human behavior is difficult. The average wireless user is simply not familiar with the threats imposed by connecting to an open Wi-Fi network at a local coffee shop or airport. In addition, users may unknowingly connect to a wireless network that they believe is the legitimate access point but that has, in fact, been set up as a honeypot or open network specifically to attract unsuspecting victims.

For example, they may have a network at home called "Linksys." As a result, their laptop may automatically connect to any other network known as "Linksys." This built-in behavior can lead to an accidental association with a malicious wireless network, more commonly referred to as wireless phishing.

Once an attacker gains access to the user's laptop, not only could the attacker pilfer information such as sensitive files, but the attacker could also harvest wireless network credentials for the user's corporate network. This attack may be far easier to perform than attacking the enterprise network directly. If an attacker can obtain the credentials from a wireless user, he or she can then use those credentials to access the corporate enterprise wireless network, bypassing any encryption or safety mechanisms employed to prevent more sophisticated attacks.

7.4.5 Client Isolation:

Users are typically the easiest target for attackers, especially when it comes to Wi-Fi. When users are associated with an access point, they can see others attempting to connect to the access point. Ideally, most users connect to the access point to obtain Internet access or access to the corporate network, but they can also fall victim to a malicious user of that same wireless network.

In addition to eavesdropping, a malicious user can also directly target other users as long as they're associated with the same access point. Specifically, once a user authenticates and associates to the access point, he or she obtains an IP address and, therefore, layer three access. Much like a wired network, the malicious wireless user is now on the same network as the other users of that access point, making them direct targets for attack.

Wireless vendors are aware of this vulnerability and have released product features to provide client isolation for guest and corporate networks. Essentially, client isolation allows people to access the Internet and other resources provided by the access point, minus the LAN capability. When securing a Wi-Fi network, isolation is a necessity. Typically the feature is disabled by default, so ensure that it's enabled across all access points.

7.5 WIRELESS NETWORK POSITIONING AND SECURE GATEWAYS

The final point to be made about wireless network hardening is related to the position of the wireless network in the overall network design topology. Owing to the peculiarities of wireless networking, described earlier in this chapter in "Radio Frequency Security Basics," wireless networks should never be directly connected to the wired LAN. Instead, they must be treated as an insecure public network connection or, in the laxest security approach, as a DMZ. Plugging an access point directly into the LAN switch is asking for trouble (even though 802.1x authentication can alleviate the problem). A secure wireless gateway with stateful or proxy firewalling capability must separate the wireless network from the wired LAN. The most common approach today is to have APs that can be connected anywhere on the LAN, but create an encrypted tunnel back to the controller and send all traffic through it before it hits the local network. The controller will run firewalling and IDS/IPS capabilities to check this traffic before it is exposed to the internal network. If the wireless network includes multiple access points across the area and roaming user access, the access points on the "wired side" must be put on the same VLAN, securely separated from the rest of the wired network. Higher-end specialized wireless gateways combine access points, firewalling, authentication, VPN concentrator, and user roaming support capabilities. The security of the gateway protecting your wireless network—even the security of the access point itself—should never be overlooked. The majority of security problems with wireless gateways, access points, and bridges stem from insecure device management implementations, including using telnet, TFTP, default SNMP community strings, and default passwords, as well as allowing gateway and access point remote administration from the wireless side of the network. Ensure that each device's security is properly audited and use wireless-specific IDS features in concert with more traditional intrusion-detection systems working above the data-link layer.

7.6 SUMMARY

Wireless security is a multilayered time and resource-consuming process, which is nevertheless essential because wireless networks are a highly prized target for attackers looking for anonymous, free Internet access and backchannel entry into otherwise securely separated networks. Wireless security encompasses wireless-specific security policy (many tips in this chapter are helpful in constructing one), radio frequency security, layer

two–specific wireless protocol security issues and solutions, higher-layer VPN and device management security, and above all, correct wireless network design with security in mind.

7.7 QUESTIONS

1) Write a short note on Wireless Vulnerabilities and Mitigations.

2) Explain Radio Frequency Security.

7.8 REFERENCE

- The Complete Reference: Information Security, Mark Rhodes-Ousley, McGrawHill, Second Edition.

*****

UNIT - IV

8

INTRUSION DETECTION AND PREVENTION SYSTEMS

Unit Structure

8.0 OBJECTIVES

After going through this unit, you will be able to:

- Understand IDS/IPS concepts
- Describe the different IDS and IPS types
- Identify features to help you evaluate different solutions
- Discuss real-life deployment considerations.
- Understanding of both systems
- Prepare to navigate the toughest operational issues.

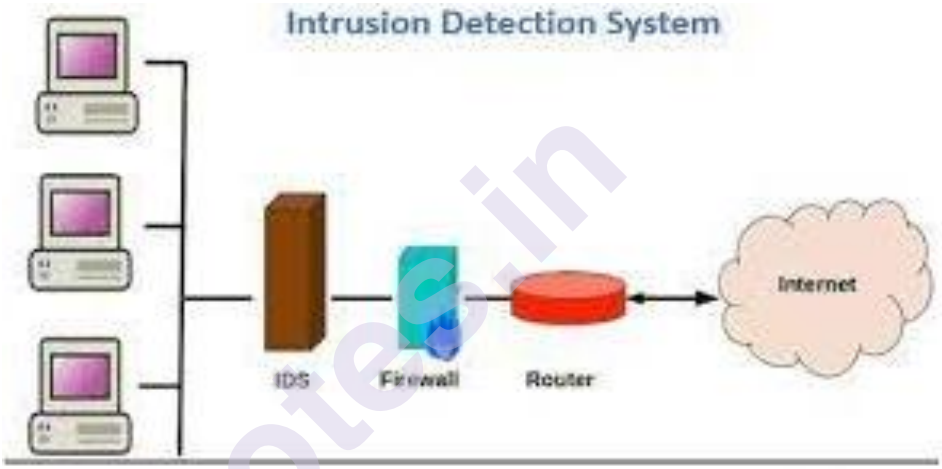
8.1 INTRODUCTION

- Intrusion detection systems (IDSs) and intrusion prevention systems (IPSs) are important tools in a computer security arsenal.
- Often thought of as a tertiary extra after antivirus software and firewalls, an IDS is often the best way to detect a security breach.
- As useful as they can be, however, successfully deploying an IDS or IPS is one of the biggest challenges a security administrator can face.

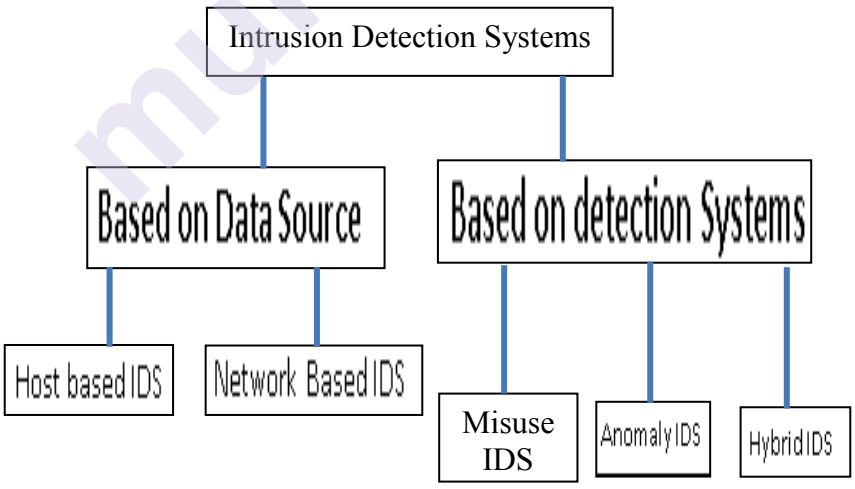
8.1.1 IDS Concepts:

- Intrusion detection (ID) is the process of monitoring and identifying specific malicious traffic.
- Most network administrators do ID all the time without realizing it.
- Security administrators are constantly checking system and security log files for something suspicious.
- An antivirus scanner is an ID system when it checks files and disks for known malware.
- Administrators use other security audit tools to look for inappropriate rights, elevated privileges, altered permissions, incorrect group memberships, unauthorized registry changes, malicious file manipulation, inactive user accounts, and unauthorized applications.
- An IDS is just another tool that can monitor host system changes (host-based) or sniff network packets of the wire (network-based) looking for signs of malicious intent.
- An IDS can take the form of a software program installed on an operating system, but today's commercial network-sniffing IDS/IPS typically takes the form of a hardware appliance because of performance requirements.
- An IDS uses either a packet-level network interface driver to intercept packet traffic or it "hooks" the operating system to insert inspection subroutines.
- An IDS is a sort of virtual food taster, deployed primarily for early detection, but increasingly used to prevent attacks.
- When the IDS notices a possible malicious threat, called an event, it logs the transaction and takes appropriate action.
- The action may simply be to continue to log, send an alert, redirect the attack, or prevent maliciousness.
- If the threat is a high risk, the IDS will alert the appropriate people.
- Alerts can be sent by e-mail, Simple Network Management Protocol (SNMP), pager, SMTP to a mobile device, or console broadcast.
- An IDS supports the defense-in-depth security principle and can be used to detect a wide range of rogue events, including but not limited to the following:
    - Impersonation attempts
    - Password cracking
    - Protocol attacks

- Buffer overflows
- Installation of rootkits
- Rogue commands
- Software vulnerability exploits
- Malicious code, like viruses, worms, and Trojans
- Illegal data manipulation
- Unauthorized file access
- Denial of service (DoS) attacks



8.1.2 IDS Types:

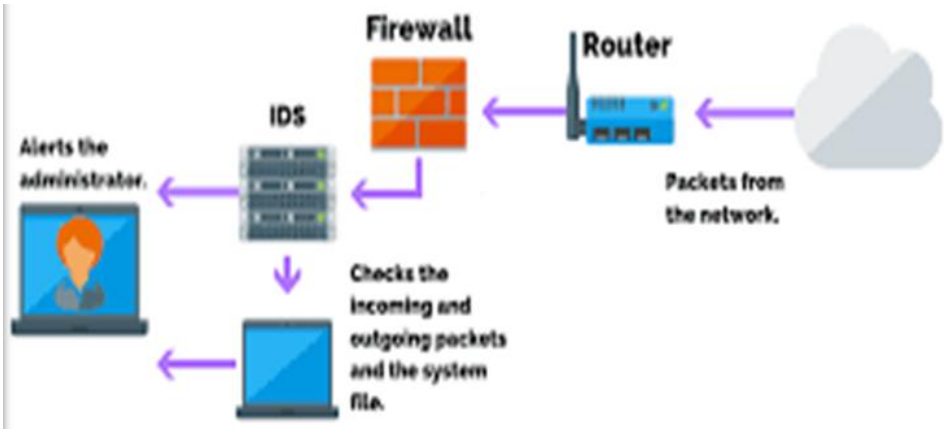


Depending on what assets you want to protect, an IDS can protect a host or a network. All IDSs follow one of two intrusion detection models—anomaly (also called profile, behavior, heuristic, or statistical) detection or signature (knowledge-based) detection - although some systems use parts of both when it's advantageous. Both anomaly and signature detection work by monitoring a wide population of events and triggering based on predefined behaviors.

1. Host-Based IDS

2. Network-Based IDS (NIDS)

1. Host-Based IDS:

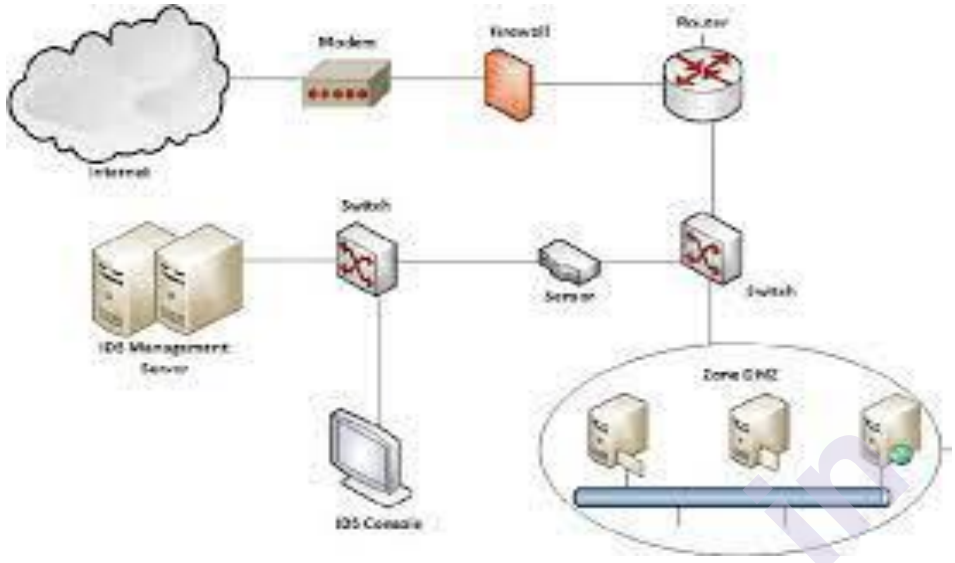


- A host-based IDS (HIDS) is installed on the host which is intended to monitor.
- The host can be a server, workstation, or any networked device (such as a printer, router, or gateway).
- HIDS installs as a service or daemon, or it modifies the underlying operating system's kernel or application to gain first inspection authority.
- Although HIDS may include the ability to sniff network traffic intended for the monitored host, it excels at monitoring and reporting direct interactions at the application layer.
- Application attacks can include memory modifications, maliciously crafted application requests, buffer overflows, or file modification attempts.
- HIDS can inspect each incoming command looking for signs of maliciousness, or simply track unauthorized file changes.
- A file-integrity HIDS (sometimes called a snapshot or checksum HIDS) takes a cryptographic hash of important files in a known clean state and then checks them again later for comparison.
- If any changes are noted, the HIDS alerts the administrator that there may be a change in integrity.
- A behavior-monitoring HIDS performs real-time monitoring and intercepts potentially malicious behavior.
- For instance, a Windows HIDS reports on attempts to modify the registry, manipulate files, access the system, change passwords, escalate privileges, and otherwise directly modify the host.

- On a Unix host, a behavior-monitoring HIDS may monitor attempts to access system binaries, attempts to download password files, and change permissions and scheduled jobs.
- A behavior-monitoring HIDS on a web server may monitor incoming requests and report maliciously crafted HTML responses, cross-site scripting attacks, or SQL injection code.
- Early warning and prevention are the greatest advantages of real-time HIDS. Because real-time HIDS is always monitoring system and application calls, it can stop potentially malicious events from happening in the first place.
- On the downside, real-time monitoring takes up significant CPU cycles, which may not be acceptable on a high-performance asset, like a popular web server or a large database server.
- Real-time behavior monitoring only screens previously defined threats, and new attack vectors are devised several times a year, meaning that real-time monitors must be updated, much like databases for an antivirus scanner.
- In addition, if an intrusion successfully gets by the real-time behavior blocker, the HIDS won't be able to provide as much detailed information about what happened thereafter as a snapshot HIDS would. Snapshot HIDSs are reactive by nature.
- They can only report maliciousness, not stop it. A snapshot HIDS excels at forensic analysis. With one report, you can capture all the changes between a known good state and the corrupted state.
- You will not have to piece together several different progressing states to see all the changes made since the baseline. Damage assessment is significantly easier than with a real-time HIDS because a Snapshot HIDS can tell you exactly what has changed.
- You can use comparative reports to decide whether you have to rebuild the host completely or whether a piecemeal restoration can be done safely. You can also use the before and after snapshots as forensic evidence in an investigation.
- Snapshot systems are useful outside the realm of computer security, too. You can use a snapshot system for configuration and change management. A snapshot can be valuable when you have to build many different systems with the same configuration settings as a master copy.
- You can configure the additional systems and use snapshot comparison to see if all configurations are identical. You can also run snapshot reports later to see if anyone has made unauthorized changes to a host.

- The obvious disadvantage of a snapshot HIDS is that alerting and reporting are done after the fact. By then, the changes have already occurred, and the damage is done.

2. Network-Based IDS (NIDS):



- Network-based IDS (NIDS) is the most popular IDS, and they work by capturing and analyzing network packets speeding by on the wire.
- Unlike HIDS, NIDS is designed to protect more than one host. It can protect a group of computer hosts, like a server farm, or monitor an entire network.
- Captured traffic is compared against protocol specifications and normal traffic trends or the packet's payload data is examined for malicious content.
- If a security threat is noted, the event is logged and an alert is generated. With HIDS, you install the software on the host you want to monitor and the software does all the work.
- Because NIDS works by examining network packet traffic, including traffic not intended for the NIDS host on the network, it has a few extra deployment considerations.
- It is common for brand-new NIDS users to spend hours wondering why their IDS isn't generating any alerts.
- Sometimes it's because there is no threat traffic to alert on, and other times it's because the NIDS isn't set up to capture packets headed to other hosts.
- A sure sign that the network layer of your NIDS is misconfigured is that it only picks up broadcast traffic and traffic headed for it specifically.

- Traffic doesn't start showing up at the NIDS simply because it was turned on.
- You must configure your NIDS and the network so the traffic you want to examine is physically passed to NIDS.
- NIDS must have promiscuous network cards with packet-level drivers, and they must be installed on each monitored network segment.
- Network taps, a dedicated appliance used to mirror a port or interface physically, and Switch Port Analysis (SPAN), are the two most common methods for setting up monitoring on a switched network.

2.1 Packet-Level Drivers:

- Network packets are captured using a packet-level software driver bound to a network interface card.
- Many Unix and Windows systems do not have native packet-level drivers built in, so IDS implementations commonly rely on open-source packet-level drivers.
- Most commercial IDSs have their own packet-level drivers and packet-sniffing software.

2.2 Promiscuous Mode:

- For NIDS to sniff packets, the packets have to be given to the packet-level driver by the network interface card.
- By default, most network cards are not promiscuous, meaning they only read packets of the wire that are intended for them.
- This typically includes unicast packets, meant solely for one particular workstation, broadcast packets meant for every computer that can listen to them, and multicast traffic meant for two or more previously defined hosts.
- Most networks contain unicast and broadcast traffic. Multicast traffic isn't as common, but it is gaining in popularity for web-streaming applications.
- By default, a network card in normal mode drops traffic destined for other computers and packets with transmission anomalies (resulting from collisions, bad cabling, and so on).
- If you are going to set up an IDS, make sure its network interface card has a promiscuous mode and is able to inspect all traffic passing by on the wire.

Sensors for Network Segments:

- For the purposes of this chapter, a network segment can be defined as a single logical packet domain.
- For NIDS, this definition means that all network traffic heading to and from all computers on the same network segment can be physically monitored.
- You should have at least one NIDS inspection device per network segment to monitor a network effectively.
- This device can be a fully operational IDS interface or, more commonly, a router or switch interface to which all network traffic is copied, known as a span port, or a traffic repeater device, known as a sensor or tap.
- One port plugs into the middle of a connection on the network segment to be monitored, and the other plugs into a cable leading to the central IDS console.
- Routers are the edge points of network segments, and you must place at least one sensor on each segment you wish to monitor.
- Most of today's networks contain switch devices. With the notable exception of broadcast packets, switches only send packets to a single destination port.
- On a switched network, an IDS will not see its neighbor's non-broadcast traffic.
- Many switches support port mirroring, also called port spanning or traffic redirection.
- Port mirroring is accomplished by instructing the switch to copy all traffic to and from a specific port to another port where the IDS sits.

8.1.3 Detection Models:

Anomaly-Detection (AD) Model:

- Anomaly detection (AD) was proposed in 1985 by noted security laureate Dr. Dorothy E. Denning, and it works by establishing accepted baselines and noting exceptional differences.
- Baselines can be established for a particular computer host or for a particular network segment.
- Some IDS vendors refer to AD systems as behavior-based since they look for deviating behaviors.
- If an IDS looks only at network packet headers for differences, it is called protocol anomaly detection.

- Several IDSs have anomaly-based detection engines. Several massively distributed AD systems monitor the overall health of the Internet, and a handful of high-risk Internet threats have been minimized over the last few years because unusual activity was noticed by a large number of correlated AD systems.
- The goal of AD is to be able to detect a wide range of malicious intrusions, including those for which no previous detection signature exists.
- By learning known good behaviors during a period of "profiling," in which an AD system identifies and stores all the normal activities that occur on a system or network, it can alert to everything else that doesn't fit the normal profile.
- Anomaly detection is statistical in nature and works on the concept of measuring the number of events happening in a given time interval for a monitored metric.
- A simple example is someone logging in with the incorrect password too many times, causing an account to be locked out and generating a message to the security log.
- Anomaly detection IDS expands the same concept to cover network traffic patterns, application events, and system utilization.

Here are some other events AD systems can monitor and trigger alerts from:

- Unusual user account activity
- Excessive file and object access
- High CPU utilization
- Inappropriate protocol use
- Unusual workstation login location
- Unusual login frequency
- A High number of concurrent logins
- A High number of sessions
- Any code manipulation
- Unexpected privileged use or escalation attempts
- Unusual content

An accepted baseline may be that network utilization on a particular segment never rises above 20 percent and routinely only includes HTTP, FTP, and SMTP traffic:

- An AD baseline might be that there are no unicast packets between workstations and only unicasts between servers and workstations.
- If a DoS attack pegs the network utilization above 20 percent for an extended period of time, or someone tries to telnet to a server on a monitored segment, the IDS would create a security event.
- Excessive repetition of identical characters in an HTTP response might be indicative of a buffer overflow attempt.
- When an AD system is installed, it monitors the host or network and creates a monitoring policy based on the learned baseline.
- The IDS or installer chooses which events to measure and how long the AD system should measure to determine a baseline.
- The installer must make sure that nothing unusual is happening during the sampling period that might skew the baseline.
- Anomalies are empirically measured as a statistically significant change from the baseline norm.
- The difference can be measured as a number, a percentage, or as a number of standard deviations.
- In some cases, like the access of an unused system file or the use of an inactive account, one instance is enough to trigger the AD system.
- For normal events with ongoing activity, two or more statistical deviations from the baseline measurement create an alert.

AD Advantages:

- AD systems are great at detecting a sudden high value for some metric. For example, when the SQL Slammer worm ate up all available CPU cycles and bandwidth on affected servers and networks within seconds of infection, you can bet AD systems went off.
- They did not need to wait until an antivirus vendor released an updated signature. As another example, if your AD system defines a buffer overflow as any traffic with over a thousand repeating characters, it will catch any buffer overflow, known or unknown that exceeds that definition.
- It doesn't need to know the character used or how the buffer overflow works.
- If your AD system knows your network usually experiences ten FTP sessions in a day, and suddenly it experiences a thousand, it will likely catch the suspicious activity.

AD Disadvantages:

- Because AD systems base their detection on deviation from what's normal, they tend to work well in static environments, such as on servers that do the same thing day in and day out, or on networks where traffic patterns are consistent throughout the day.
- On more dynamic systems and networks that, therefore, have a wider range of normal behaviors, false positives can occur when the AD triggers something that wasn't captured during the profiling period.

Signature-Detection Model:

- A Signature-detection or misuse IDS is the most popular type of IDS, and they work by using databases of known bad behaviors and patterns.
- This is nearly the exact opposite of AD systems.
- When you think of a signature-detection IDS, think of it as an antivirus scanner for network traffic.
- Signature-detection engines can query any portion of a network packet or look for a specific series of data bytes.
- The defined patterns of code are called signatures, and often they are included as part of a governing rule when used within an IDS.
- Signatures are byte sequences that are unique to a particular malady.
- A byte signature may contain a sample of virus code, a malicious combination of keystrokes used in a buffer overflow, or text that indicates the attacker is looking for the presence of a particular file in a particular directory.
- For performance reasons, the signature must be crafted so it is the shortest possible sequence of bytes needed to detect its related threat reliably.
- It must be highly accurate in detecting the threat and not cause false positives.
- Signatures and rules can be collected together into larger sets called signature databases or rule sets.

Signature-Detection Rules:

- Rules are the heart of any signature-detection engine.
- A rule usually contains the following information as a bare minimum:
  - Unique signature byte sequence
  - Protocol to examine (such as TCP, UDP, ICMP)

- IP port requested
- IP addresses to inspect (destination and source)
- Action to take if a threat is detected (such as allow, deny, alert, log, disconnect)

- Most IDSs come with hundreds of predefined signatures and rules. They are either all turned on automatically or you can pick and choose.
- Each activated rule or signature adds processing time for analyzing each event.
- If you were to turn on every rule and inspection option of a signature-detection IDS, you would likely find it couldn't keep up with traffic inspection.
- Administrators should activate the rules and options with an acceptable cost/benefit tradeoff.
- Most IDSs also allow you to make custom rules and signatures, which is essential for responding immediately to new threats or for fine-tuning an IDS.
- Here are some hints when creating rules and signatures:
  - Byte signatures should be as short as possible, but reliable, and they should not cause false positives.
  - Similar rules should be near each other. Organizing your rules speeds up future maintenance tasks.
  - Some IDSs and firewalls require rules that block traffic to appear before rules that allow traffic. Check with your vendor to see if rule placement matters.
  - Create wide-sweeping rules that do the quickest filtering first. For example, if a network packet has a protocol anomaly, it should cause an alert event without the packet ever getting to the more processor-intensive content scanning.
  - To minimize false positives, rules should be as specific as possible, including information that specifically narrows down the population of acceptable packets to be inspected.
- Some threats, like polymorphic viruses or multiple-vector worms, require multiple signatures to identify the same threat. For instance, many computer worms arrive as infected executables, spread over internal drive shares, send themselves out with their own SMTP engines, drop other Trojans and viruses, and use Internet chat channels to spread.
- Each attack vector would require a different signature.

Advantages of Signature Detection:

- A Signature-detection IDS is proficient at recognizing known threats.
- Once a good signature is created, signature detection IDS is great at finding patterns, and because they are popular, a signature to catch a new popular attack usually exists within hours of it first being reported.
- This applies to most open-source and commercial vendors. Another advantage of a signature-detection IDS is that it will specifically identify the threat, whereas an AD engine can only point out a generality.
- An AD IDS might alert you that a new TCP port opened on your file server, but a signature-detection IDS will tell you what exploit was used.
- Because a signature-detection engine can better identify specific threats, it has a better chance of providing the correct countermeasure for intrusion prevention.

Disadvantages of Signature Detection:

- Although signature-detection IDS is the most popular type of IDS, they have several disadvantages as compared to an AD IDS.

Cannot Recognize Unknown Attacks:

- Just like antivirus scanners, signature-detection IDS is not able to recognize previously unknown attacks.
- Attackers can change one byte in the malware program (creating a variant) to invalidate an entire signature.
- Hundreds of new malware threats are created every year, and signature-based IDS is always playing catch up.
- To be fair, there hasn't been a significant threat in the last few years that didn't have a signature identified by the next day, but your exposure is increased in the so-called zero hours.

Performance Suffers as Signatures or Rules Grow:

- Because each network packet or event is compared against the signature database or at least a subset of the signature database, performance suffers as rules increase.
- Most IDS administrators using signature detection usually end up only using the most common signatures and not the less common rules.
- The more helpful vendors rank the different rules with threat risks so the administrator can make an informed risk tradeoff decision.

- Although this is an efficient use of processing cycles, it does decrease detection reliability.
- Because a signature is a small, unique series of bytes, all a threat coder has to do is change one byte that is identified in the signature to make the threat undetectable.
- Threats with small changes like these are called variants. Luckily, most variants share some common portion of code that is still unique to the whole class of threats, so that one appropriate signature, or the use of wildcards, can identify the whole family.

What Type of IDS Should You Use?

- There are dozens of IDS to choose from.
- The first thing you need to do is survey the computer assets you want to protect and identify the most valuable computer assets that should get a higher level of security assurance.
- These devices are usually the easiest ones to use when making an ROI case to management.
- New IDS administrators should start small, learn, fine-tune, and then grow.
- A HIDS should be used when you want to protect a specific valuable host asset.

8.1.4 IDS Features:

- A NIDS should be used for general network awareness and as an early warning detector across multiple hosts.
- You need to pick an IDS that supports your network topology, operating system platforms, budget, and experience.
- If you have a significant amount of wireless traffic exposed in public areas, consider investing in a wireless IPS.
- If you have high-speed links that you need to monitor, make sure your IDS has been rated and tested at the same traffic levels.
- IDS should be based on a product that does both, anomaly, and signature detection.
- The best IDS utilizes all techniques, combining the strengths of each type to provide a greater defense strategy.
    - IDS is more than detection engines.
    - Detection is their main purpose, but if you can't configure the system or get the appropriate information out of the IDS, it won't be much helpful.

IDS End-User Interfaces:

- IDS end-user interfaces let you configure the product and see ongoing detection activities.
- You should be able to configure operational parameters, rules, alert events, actions, log files, and update mechanisms.
- IDS interfaces come in two flavors: syntactically difficult command prompts or less-functional GUIs.
- Historically, IDS is a command-line beast with user-configurable text files.
- Command line consoles are available on the host computer or can be obtained by a Telnet session or proprietary administrative software.
- The configuration files control the operation of the IDS detection engine, define and hold the detection rules, and contain the log files and alerts.
- You configure the files, save them, and then run the IDS.
- If any runtime errors appear, you have to reconfigure and rerun.
- A few of the command-line IDS programs have spawned GUI consoles that hide the command-line complexities.
- Although text-based user interfaces may be fast and configurable, they aren't loved by the masses.
- Hence, more and more IDS are coming up with user-friendly GUIs that make installation a breeze and configuration a matter of point-and-click.
- With few exceptions, the GUIs tend to be less customizable than their text-based cousins and, if connected to the detection engine in real time can cause slowness.
- Many of the GUI consoles present a pretty picture to the end user but end up writing settings to text files, so you get the benefits of both worlds.

Intrusion-Prevention Systems (IPS):

- Since the beginning, IDS developers have wanted the IDS to do more than just monitor and report maliciousness.
- What good is a device that only tells you you've been maligned when the real value is in preventing the intrusion? That's like a car alarm telling you that your car has been stolen, after the fact.
- Like intrusion detection, intrusion prevention has long been practiced by network administrators as a daily part of their routine.

- Setting access controls, requiring passwords, enabling real-time antivirus scanning, updating patches, and installing perimeter firewalls are all examples of common intrusion-prevention controls.
- Intrusion-prevention controls, as they apply to IDSs, involve real-time countermeasures taken against a specific, active threat. For example, the IDS might notice a ping flood and deny all future traffic originating from the same IP address.
- Alternatively, a host-based IDS might stop a malicious program from modifying system files.
- Going far beyond mere monitoring and alerting, second-generation IDS is being called an intrusion-prevention system (IPS).
- They either stop the attack or interact with an external system to put down the threat.
- If the IPS, as shown in Figure A, is a mandatory inspection point with the ability to filter real-time traffic, it is considered inline.
- Inline IPS can drop packets, reset connections, and route suspicious traffic to quarantined areas for inspection.
- If the IPS isn't in line and is only inspecting the traffic, it still can instruct other network perimeter systems to stop an exploit.
- It may do this by sending scripted commands to a firewall, instructing it to deny all traffic from the remote attacker's IP address, calling a virus scanner to clean a malicious file, or simply telling the monitored host to deny the hacker's intended modification.
- For an IPS to cooperate with an external device, they must share a common scripting language, API, or some other communicating mechanism.
- Another common IPS method is for the IDS device to send reset (RST) packets to both sides of the connection, forcing both source and destination hosts to drop the communication.
- This method isn't seen as being very accurate, because often the successful exploit has happened by the time a forced reset has occurred, and the sensors themselves can get in the way and drop the RST packets.

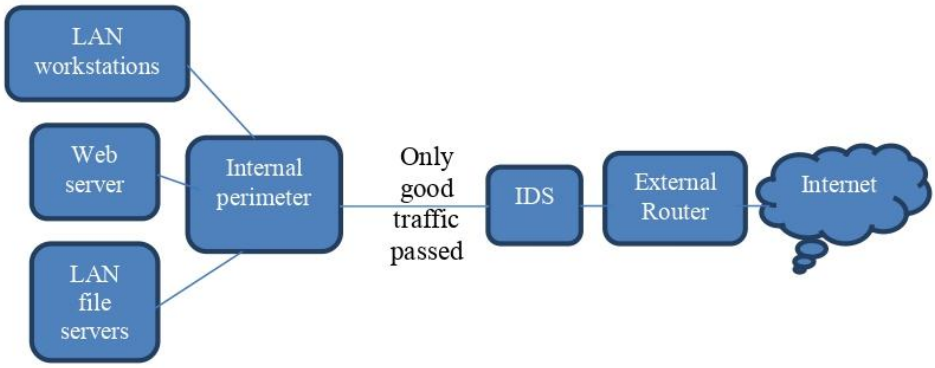
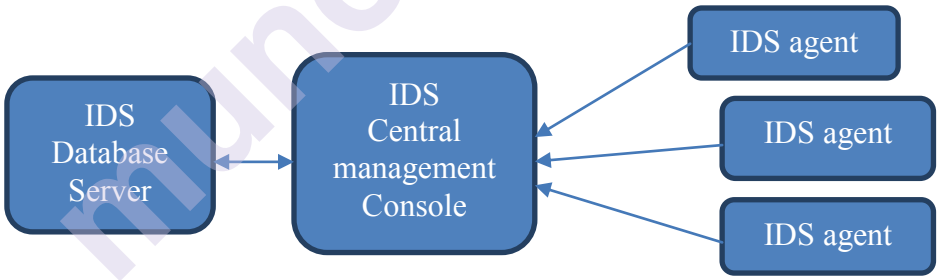


Figure A

IPS Disadvantages:

- A well-known consequence of IPS is its ability to exacerbate the effects of a false positive.
- With an IDS, a false positive leads to wasted log space and time, as the administrator researches the threat's legitimacy.
- IPS is proactive, and a false positive means a legitimate service or host is being denied.
- Malicious attackers have even used prevention countermeasures such as DoS attack.

IDS Management:



- Central to the IDS field are the definitions of management console and agent.
- An IDS agent (which can be a probe, sensor, or tap) is the software process or device that does the actual data collection and inspection.
- If you plan to monitor more than two network segments, you can separately manage multiple sensors by connecting them to a central management console.
- This allows you to concentrate your IDS expertise at one location. IDS management consoles usually fulfill two central roles:
    - Configuration and reporting. If you have multiple agents, a central console can configure and update multiple distributed agents at once. For example, if you discover a new type of attack, you can

use the central console to update the attack definitions for all sensors at the same time.

- A central console also aids in determining agent status—active and online or otherwise.
- In environments with more than one IDS agent, reporting captured events to a central console is crucial.
- This is known as event aggregation. If the central console attempts to organize seemingly distinct multiple events into a smaller subset of related attacks, it is known as event correlation.
- For example, if a remote intruder port scans five different hosts, each running its own sensor, a central console can combine the events into one larger event.
- To aid in this type of correlation analysis, most consoles allow you to sort events by:
  - Destination IP address
  - Source IP address
  - Type of attack
  - Type of protocol
  - Time of attack
- You can also customize the policy that determines whether two separate events are related. For example, you can tell the console to link all IP fragmentation attacks in the last five minutes into one event, no matter how many source IP addresses were involved.
- Agents are configured to report events to the central console, and then the console handles the job of alerting system administrators.
- This centralization of duties helps with setting useful alert thresholds and specifying who should be alerted.
- Changes to the alert notification list can be made on one computer instead of on numerous distributed agents.
- A management console can also play the role of an expert analyzer.
- A lightweight IDS performs the role of agent and analyzer on one machine.
- In larger environments with many distributed probes, agents collect data and send it to the central console without determining whether the monitored event was malicious or not.
- The central console manages the database warehousing for all the collected event data.

- As shown in Figure B, the database may be maintained on a separate computer connected with a fast link.

8.1.5 IDS Deployment Considerations:

- IDS is a beneficial tool, but it has weaknesses.
- They need to be fine-tuned if you want to maximize their usefulness, and if you intend to deploy one, you'll need to come up with a deployment plan to do so successfully.
- Creating this usually represents a substantial amount of work.
- This section summarizes these deployment issues.

IDS Fine-Tuning:

- Fine-tuning an IDS means doing three things:
  - Increasing inspection speed
  - Decreasing false positives
  - Using efficient logging and alerting.

Increasing Inspection Speed:

- Most IDS administrators start monitoring all packets and capturing full packet decodes.
- You can narrow down what packets an IDS inspects by telling it to include or ignore packets based on source and destination addresses.
- For example, if you are most concerned with protecting your servers, modify the IDS packet inspection engine so it only captures packets with server destination addresses.
- Another common packet filter is a rule that excludes broadcast packets between routers.
- Routers are always busy chatting and broadcasting to learn routes and reconstruct routing tables, but if you aren't worried about internal ARP poisoning, don't capture ARP packets.
- The more packets the IDS can safely ignore, the faster it will be.
- Another strategy is to let other faster perimeter devices do the filtering.
- Routers and firewalls are usually faster than IDS, so when possible, configure the packet filters of your routers and firewalls to deny traffic that should not be on your network in the first place.
- For example, tell your router to deny IP address spoofs, and tell your firewall to drop all NetBIOS traffic originating from the Internet.

- The more traffic that you can block with the faster device, the higher-performing your IDS will be.
- That's the way it should be—each security device should be configured to excel at what it does best, at the layer from which it does it best.

Decreasing False Positives:

- Because IDS have so many false positives, the number one job of any IDS administrator is to track down and troubleshoot false positives.
- In most instances, false positives will outweigh all other events.
- Track them all down, rule out maliciousness, and then appropriately modify the source or IDS to prevent them.
- Often the source of the false positive is a misbehaving program or a chatty router.
- If you can't stop the source of the false positive, modify the IDS so it will not track the event.
- The key is that you want your logs to be as accurate as they can be, and they should only alert you to events that need human intervention.
    - Don't get into the habit of ignoring the frequently occurring false positives in your logs as a way of doing business.
    - This will quickly lead to your missing the real events buried inside all the false positives—or to the logs not being read at all.

Using Efficient Logging and Alerting:

- Most vendor products come with their own preset levels of event criticalities, but when setting up the IDS, take the time to customize the criticalities for your environment.
- For instance, if you don't have any Apache web servers, set Apache exploit notices with a low level of prioritization. Better yet, don't track or log them at all.

IPS Deployment Plan:

- So you want to deploy your first IPS.
- You've mapped your network, surveyed your needs, decided what to protect, and picked an IPS solution.
- Here are the steps to a successful IPS deployment:
    - Document your environment's security policy.
    - Define human roles.
    - Decide the physical location of the IPS and sensors.

- Configure the IPS sensors and management console to support your security policy.
- Plan and configure device management (including the updated policy).
- Review and customize your detection mechanisms.
- Plan and configure any prevention mechanisms.
- Plan and configure your logging, alerting, and reporting.
- Deploy the sensors and console (do not encrypt communication between sensors and links to lessen troubleshooting).
- Test the deployment using IPS testing tools (initially use very broad rules to make sure the sensors are working).
- Encrypt communications between the sensors and console.
- Test the IPS setup with actual rules.
- Analyze the results and troubleshoot any deficiencies.
- Fine-tune the sensors, console, logging, alerting, and reporting.
- Implement the IPS system in the live environment in monitor-only mode.
- Validate alerts generated from the IPS.
- One at a time, set blocking rules for known reliable alerts that are important in your environment.
- Continue adding blocking rules over time as your confidence in each rule increases.
- Define continuing education plans for the IPS administrator.
- Repeat these steps as necessary over the life of the IPS.

- Installing and testing an IPS is a lot of work.
- The key is to take small steps in your deployment and plan and configure all the parts of your IPS before just turning it on.
- The more time you spend on defining reporting and database mechanisms at the beginning, the better the deployment will go.
- Use a test rule that is sure to trigger the IPS sensor or console on every packet.
- This ensures that the physical part of the sensor is working and lets you test the logging and alerting mechanisms.

- Once you know the physical layer is working, you can remove that test rule (or comment it out or unselect it, in case you need it later).
- Do not turn on encryption, digital signing, or any other self-securing components until after you've tested the initial physical connections.
- This reduces troubleshooting time caused by mistyped passphrases or incorrectly configured security settings.
- Finally, keep on top of your logs, and research all critical events.
- Quickly rule out false positives, and fine-tune your IPS on a regular basis to minimize false positives and false negatives.
- Once you get behind in your log duty, catching up again is tough.
- Successful IPS administrators track and troubleshoot everything as quickly as they can.
- The extra effort will pay dividends with smaller and more accurate logs.

8.1.6 Security Information and Event Management (SIEM):

- Multiple security systems can report to a centralized Security Information and Event Management (SIEM) system, bringing together logs and alerts from several disparate sources.
- You may find different combinations of references to the acronym SIEM, owing to the evolution of capabilities and the consequent variety of names attached to SIEM products over the years, such as "Security Incident and Event Management" or "Security Incident and Event Monitoring."
- These are all the same thing—a technology to collect, analyze, and correlate events and alerts generated by monitoring systems.
- SIEM platforms take the log files, find commonalities (such as attack types and threat origination), and summarize the results for a particular time period.
- For example, all logs and alerts from all IDSs, perimeter firewalls, personal firewalls, antivirus scanners, and operating systems can be tied together.
- Events from all logs are then gathered, analyzed, and reported on from one location.
- SIEMs offer the ultimate in-event correlation, giving you one place to get a quick snapshot of your system's security or to get trend information.
- SIEMs can also coordinate signature and product updates.

- SIEMs have a huge advantage over individual IDS systems because they have the capability to collect and analyze many different sources of information to determine what's really happening.
- As a result, the SIEM can significantly reduce false positives by verifying information based on other data.
- That data comes from many sources, including workstations, servers, computing infrastructure, databases, applications, network devices, and security systems.
- Because all those sources generate a vast amount of real-time data, SIEM products need to be fast and effective, with a significant amount of storage and computing power.
- Today's network attacks are often complex—slow, multifaceted, and stealthy.
- Attackers use many techniques to circumvent security controls.
- Slow attacks can spread malicious network traffic over days, weeks, or even months, hiding inside the massive data streams experienced on any given network.
- Multifaceted attacks use a variety of techniques in the hope that at least one will succeed, or that the distributed nature of the attacks will distract attention away from the source.
- Stealthy attacks use obscure or nonstandard aspects of network technologies and protocols to slip past traditional monitoring capabilities that have been programmed based on the assumption that network traffic will always follow normal standards.
- An IDS needs SIEM to detect these advanced attacks.
- SIEM is one of the most important tools used by security operations and monitoring staff because it provides a one-stop visibility processing environment and attacks against those areas.
- Let's take a look at what SIEM can do:-

Data Aggregation:

- SIEM collects information from every available source that is relevant to a security event.
- These sources take the form of alerts, real-time data, logs, and supporting data.
- Together, these provide the correlation engine of the SIEM with information it can use to make decisions about what to bring to the security administrator's attention.

- Consider the following examples of specific data sources consumed by SIEM.

Alerts:

- When is an alert real, and when is it a false positive? This is the key question associated with an IDS and a source of frustration for security administrators in charge of tuning IDS.
- This is where SIEM enters into the picture.
- The key function of SIEM is to validate security alerts using many different sources of data to reduce false positives, so only the most reliable alerts get sent on to the security administrator.
- Thus, the alerts from all IDS sources as well as all other security monitoring systems should be given only to the SIEM, so it can decide which ones to pass along.

Real-Time Data:

- Real-time data such as network flow data (for instance, Cisco's NetFlow and similar traffic monitoring protocols from other vendors) gives the SIEM additional information to correlate.
- Streaming this data into the SIEM provides important information about normal and abnormal traffic patterns that can be used in conjunction with alerts to determine whether an attack is in progress.
- For example, an unusually high amount of SMTP traffic that accompanies several malware alerts may result in a high confidence alert that an e-mail worm is on the loose.
- Similarly, an abnormally high amount of inbound Internet traffic, combined with a high number of firewall deny events, can indicate a denial of service attack.
- Another example is fragmented or truncated network packets, which may indicate a network-based attack.
- Each of these real-time data elements gives the SIEM important validation data for IDS alerts.

Logs:

- Logs are different from events, in that they are a normal part of system activity and are usually meant for debugging purposes.
- Logs can be an important additional data source for a SIEM, however.
- Logs contain valuable information about what's happening on a system, and they can give the SIEM a deeper view of what's happening.

- For example, login failures that may otherwise go unnoticed by a system administrator because they are buried in a system log might be of great interest to SIEM, especially if there are many login failures for a single account (indicating a possible focused attempt to break into that account) or, similarly, if there are login failures on many different accounts, which may indicate a broad-based attempt to break into accounts using common passwords.
- System errors that are logged and collected by SIEM are a valuable source of correlating information.
- In addition to providing the SIEM itself with detailed information, logs can be used to make decisions about the validity of IDS alerts and they are easier for humans to view in SIEM.
- The system administrator who needs to find a particular log entry may find SIEM is the best option for searching and finding that log entry.

Ideal log sources for any SIEM include the following:

- End-user computers
- Windows and Unix servers
- Domain controllers
- DNS and DHCP servers
- Mail servers
- Databases
- Web servers
- Applications
- Switches and routers
- VPN concentrators
- Firewalls
- Web filters and proxies
- Antivirus

- Logs can be sent to the SIEM in a couple of different ways: they can be pushed to the SIEM by the individual devices that collect the logs, or they can be pulled in by the SIEM itself.
- The Syslog protocol, which is widely used by Unix systems as well as network devices, is an example of a push technique.
- When the IP address of the SIEM is configured in the Syslog service of a server or device, each log entry that the device produces will be sent over the network to the SIEM.

- For systems that don't support Syslog, such as Windows, third-party software can be used to collect static log information and send it to the SIEM.
- The third-party software agent can be installed directly on the reporting server, or on a central server built for log collection, in which case the software periodically connects to the server, grabs the latest log entries, and pushes them to the SIEM.
- Whether pushed or pulled, log entries need to be parsed.
- Every vendor has a different format for the fields in their Syslog data.
- Even though they all use the same protocol, the information contained within the log is not standardized.
- Modern SIEM products come with dozens of parsers that have been preconfigured to convert the Syslog fields of different manufacturers into a format the SIEM can use.
- In the rare cases where a built-in parser is not available for a particular vendor's Syslog format, the SIEM allows the administrator to define a custom mapping.

Supporting Data:

- You can enhance the quality of a SIEM's correlation even more by providing the SIEM with supporting data that has been previously collected.
- Data can be imported into the SIEM, and it will use that data to make comparative determinations.
- For example, asset management data containing names, IP addresses, operating systems, and software versions gives the SIEM valuable information it can use to determine whether an IDS alert makes sense within the context of the software environment.
- Coupled with risk weighting data, the SIEM can use this information to prioritize and escalate alerts that pertain to high-risk systems.
- You can also use vulnerability scans to give the SIEM information it can use to compare an alert about an exploit with an associated vulnerability to determine if the exploit is real and whether it was successful.
- Moreover, geolocation information can be used to prioritize alerts from high-risk countries or even local areas such as the data center or public hotspots in which mobile devices might be attacked.

Analysis:

- A SIEM takes all the data given to it and makes decisions, so the security administrator can focus on the most important alerts. For this reason, event correlation is SIEM's most important feature.
- The correlation engine of every SIEM product is its most distinguishing feature.
- The better the analysis, the cleaner the end result.
- In effect, a SIEM is a sort of artificial intelligence system, working much like the human brain in putting together different elements that individually may not be important, but taken together form a picture of a critical security situation.
- A SIEM does this at a much faster rate than any human possibly could, giving the security administrator a time advantage so he or she can react quickly to attacks in progress.
- Real-time analysis of security events is only made possible with a SIEM.
- Thousands or even millions of events occur every second across most networks.
- No human can hope to see, absorb, and understand all of them at once.
- By comparison, forensic investigations in which the investigator looks at a few different data sources to decide who did what and when often take weeks of intense, focused effort.
- That's too long a timeframe for an effective response to an attack.
- To stop an attack in progress, real-time analysis is required.
- Because it collects so much data from across the enterprise, a SIEM can do more than alert.
- It can provide system administrators and network administrators with advanced search capabilities which they will not find on any other platform.
- For this reason, the SIEM represents an excellent shared platform that can make every administrator's job easier and more efficient.
- Thus, the SIEM is not just a security tool; it's also a valuable IT management tool.
- The SIEM can also perform historical and forensic analysis based on the log information it collects.

- Depending on how much storage is allocated to the SIEM, either on-board or over the network, it can retain logs and alerts for a long enough period of time that it can investigate past events.

- Security investigators can dig into the logs to find out what happened in a prior situation, and system administrators can look at past events to troubleshoot and evaluate functional issues.

8.2 LET US SUM UP

- We have seen an intrusion detection system should be a part of every network security administrator's protection plan.

- An IDS provides the "detection" aspect of the three Ds of security.

- Along with other ID tools and methods, an IDS can monitor a host for system changes or sniff network packets of the wire, looking for malicious intent.

- NIDS uses the same technology to make decisions about blocking network traffic.

- An IDS in blocking mode is known as IPS.

- A HIPS would be appropriate on strategically valuable hosts, an IDS across the network for general early-warning detection, and an IPS for critical networks that need active protection.

- SIEM systems greatly enhance the accuracy, effectiveness, and completeness of IDS alerts.

8.3 QUESTIONS

1. Explain the concept of IDS.
2. Explain different IDS types based on the network.
3. What are the different IDS detection methods? Explain.
4. Write a note on SIEM.
5. Explain different considerations of IDS deployment.
6. Explain the different features of IDS.

8.4 BIBLIOGRAPHY

- Carter, Earl, and Jonathan Hogue. Intrusion Prevention Fundamentals. Cisco Press, 2006.

- The Complete Reference: Information Security, Mark Rhodes-Ousley

8.5 REFERENCES

- https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-94.pdf
- https://www.emerald.com/insight/content/doi/10.1108/09685221011079199/full/html
- https://www.scirp.org/html/3823.html

*****

9

VOICE OVER IP (VOIP) AND PBX SECURITY

Unit Structure

9.0 OBJECTIVES

After going through this unit, you will be able to:

- Understand the security of enterprise voice, telephony, and streaming multimedia systems such as video conferencing, webcast, and multicast systems.
- Study practical approaches to both VoIP and non-VoIP telephony system security.
- Understand best practices for protecting voice communications.
- Focus on the various components of modern telecommunication infrastructure.
- Understand best practices for securing each of those components.

9.1 INTRODUCTION

Attackers have been targeting computing systems for the last 25 years or so using intentionally exploitative behavior such as hacking and denial of service attacks.

However, telephony exploits (originally referred to as phone phreaking but now included as part of mainstream hacking) have been used by clever individuals and organizations as far back as the 1960s to do everything from gaining free long distance to secretly passing malicious data right under the sensorial noses of otherwise diligent security systems.

One of the most practical approaches to VoIP and non-VoIP telephony system security is making yourself the least attractive target. In modern telecommunication infrastructures, many protocols are used, and nearly all of them cross over onto the data communication network. There is no longer a strict delineation between voice and data, and as a result, the risks to both data networks and voice networks consist of a superset of the risks to each.

9.1.1 Background:

When a VoIP system is layered on the top of an IP network, risks are associated with both which are defined as:

- Many VoIP systems are server-based and rely on common operating systems (mainly Windows and Linux) to run their hardware interface. Therefore, they are susceptible to a class of problems that from a voice systems perspective were not previously a threat.
- While providing low-cost, advanced end-user features and reliable transport mechanisms for voice traffic, IP-based voice protocols also give attackers a new method for exploiting voice systems and additional avenues for compromising data networks in general.

The following are the components of a modern enterprise IP-based phone or video system:

Call control elements (call agents):

- Appliance or server-based call control—Internet protocol private branch exchange (IPPBX)
- Soft switches
- Session border controllers (SBCs)
- Proxies

Gateways and gatekeepers:

- Dial peers
- Multi-conference units (MCUs) and specialized conference bridges

Hardware endpoints:

- Phones
- Video codecs
- Other devices and specialized endpoints

Soft clients and software endpoints:

- IP phones

- Unified messaging (UM) integrated chat and voice clients
- Desktop video clients
- IP-based smartphone clients

Contact center components:

- Automated call distribution (ACD) and interactive voice response (IVR) systems
- Call center integrations and outbound dialers
- Call recording systems
- Call center workflow solutions

Voicemail systems:

List of protocols commonly used on enterprise networks, the PTN, and the Internet:

- H.248 (also known as Megaco)
- Media gateway control protocol (MGCP)
- Session initiation protocol (SIP)
- H.323
- The Skinny call control protocol (SCCP) and other proprietary protocols
- Session description protocol (SDP), real-time protocol (RTP), real-time control protocol (RTCP), and real-time streaming protocol (RTSP)
- Secure real-time transport protocol (SRTP)
- Inter-Asterisk eXchange protocols (IAX and IAX2)
- T.38 and T.125
- Integrated services digital network (ISDN)
- Signaling system number seven (SS7) and SIGTRAN
- Short message service (SMS)

During traditional carrier networks, switches were introduced by a class with five different roles as U.S.-centric standards, these classes are:

Class 1:

International gateways handing off and receiving traffic from outside the U.S and Canadian networks

Class 2:

Tandem switches interconnecting whole regions

Class 3:

Tandem switches connecting major population centers within a region

Class 4:

Tandem switches connecting the various areas of a city or town in a region

Class 5:

Switches connecting subscribers and end-users

Anything below this level was considered a PBX or key system which effectively controls toll centers and long distances, but limited the availability of extended features such as least-cost routing.

The portability of IP and flexibility of VoIP have allowed enterprises to provide their own transport across significant geographical distances, as they are no longer relegated to the functions and features of PBX.

The main drivers of VoIP technology are the opportunities for cost savings, from lowering the cost of structured cabling by sharing Ethernet connections to advanced features like VoIP backhaul and global tail-end hop-off.

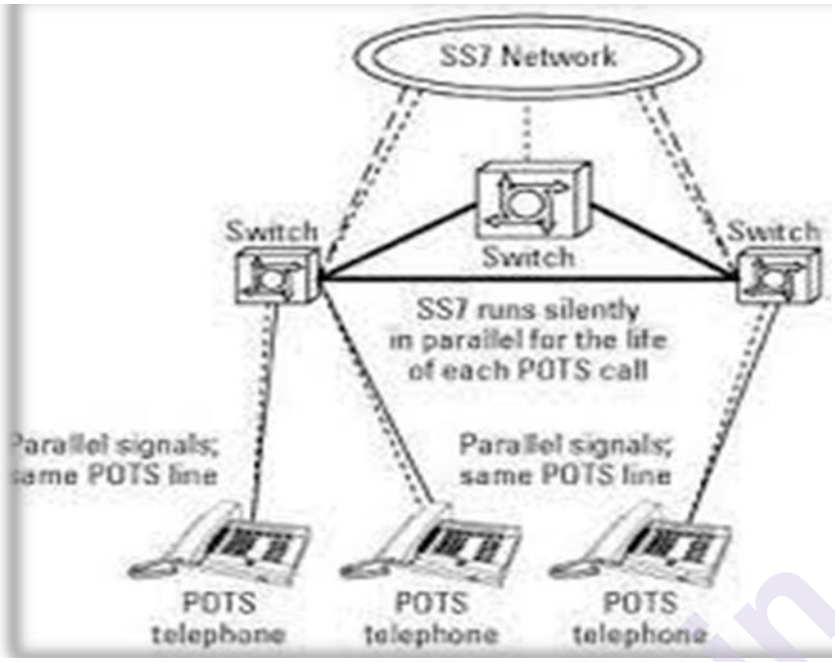
9.1.2 VoIP Components:

The major components of a VoIP network, while different in approach, deliver very similar functionality to that of a PSTN and enable VoIP networks to perform all of the same tasks that the PSTN does. The one additional requirement is that VoIP networks must contain a gateway component that enables VoIP calls to be sent to a PSTN, and vice versa.

There are four major components to a VoIP network:

- Call processing Server / IP PBX
- Voice and Media Gateways and Gatekeepers
- MCUs
- Media / VoIP Gateways
- IP network

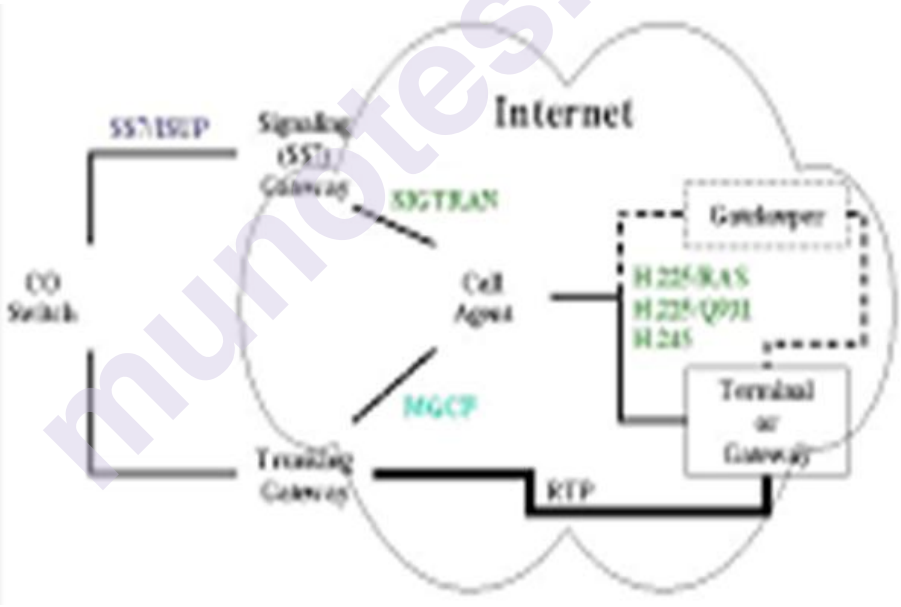
Call processing Server / IP PBX/ Call Control:



- The call control element (the "brains" of the operation) of a VoIP system can be either a purposed appliance, a piece of software that runs on a common or specialized server operating system, or a piece of network hardware embedded or integrated into another networking component such as a switchblade or software module (soft switch).
- The enterprise's original IP phone systems were traditional digital time-division multiplexing (TDM) systems with an IP-enabled component, designed like digital systems.
- They eventually evolved into full IP-based systems (IPPBX).
- They have now evolved far beyond the early designs that mimicked the "old thinking" of voice networks by leveraging the tools and resiliency available in IP networking, high-availability server architecture, and virtualization.
- Primarily responsible for call setup and teardown, signaling, device software serving, and feature configuration, call control is one of the easier pieces of the voice infrastructure to protect.
- This does not mean that security for this component should be taken lightly.
- Call control is critical to the infrastructure, particularly if any part of your business's revenue is dependent on phone calls (customer service, call centers, etc.).
- If your shop runs an IP phone system that you manage internally, this hardware sits well within your physical and logical security perimeter and should be relatively straightforward to secure.

- Following best practices related to patching, backup, and configuration management is paramount, but as long as this component is not exposed to the outside world, it is a difficult target to all but internal threats.
- There are special types of call control elements such as session border controllers (SBCs) and voice proxies that are designed to be exposed to or interface with systems under a different administrative domain.
- SBCs can also perform functions frequently required by regulations such as emergency call prioritization and lawful intercept.
- It would be wise to use one of these and to ensure they are hardened, particularly if you allow VoIP-to-PSTN calls.
- Network access control lists (ACLs) and firewalls can be employed to help and protect these and other elements of the voice infrastructure that must be exposed, and many advanced stateful firewalls now have built-in application-level gateway (ALG) capabilities designed specifically for voice protocols.

Voice and Media Gateways and Gatekeepers:



- The voice (or media) gateway is the pathway to the outside world.
- This component is what allows termination to a PSTN, transcoding between TDM and IP networks, media termination, and other types of analog/digital / IP interfaces required in today's multimedia-rich IP infrastructures.
- Gateways are configured to use dial peers (defined as "addressable endpoints") to originate and receive calls.

- Some gateways are directly managed by the call control elements via a control protocol (MGCP or H.248), whereas others operate in a more independent, stand-alone capacity (H.323 or SIP).
- Voice gateways can also run soft switches and perform primary (or survivable) call processing or "all-in-one" functions, an approach commonly used in the SMB space.
- The critical piece to consider about voice gateways is that, in stark contrast to the call control components, the gateways are nearly always exposed to the outside world in some way.
- Although not universally true based on the specific application, in an enterprise, voice gateways are the termination points for the PSTN and, as such, need to be carefully protected.
- Always ensure strong authentication methods are used to access the device itself and pay special attention to disabling unneeded services on a gateway, especially H.323 and SIP if they are not being used.
- Some systems have these protocols enabled by default, which is a recipe for disaster if they are exposed unprotected to the Internet. For example, even if you are not running SIP on your network, a voice gateway with an Internet connection, a PSTN connection, and SIP.
- Gatekeepers, not to be confused with gateways, provide intelligence and control certain routing and authentication, authorization, and accounting (AAA) security functions.
- They can also perform and assist with certain types of address translation and can consolidate administrative control elements such as call detail records (CDR), communication with monitoring and management systems, and bandwidth management for a given zone.

MCUs:

- Conferencing and collaboration are used extensively within and across all enterprises as part of the fundamental communications capability that connects all users to each other.
- At the heart of this technology is the Conference Bridge, or multi-conference unit (MCU), a multiport bridging system for audio, video, and multimedia collaboration.
- The trend between internally hosted MCUs and provider-hosted MCUs has been stuck in the yoyo of corporate decision-making, with each specific situation warranting one direction or the other based on the cost to own, cost to operate, features, and security.
- Special attention should be paid to MCU functionality, whether they are hosted on the premise or externally, in order to make sure they are secure.

Consider the following:

- The easier it is to use, the more people will use it—even the ones you don't want to use it.
- Convenience and ease of use need to be balanced with secure practices.
- A problem with MCU can affect a lot of users at once.
- MCUs can connect different types of media; require those facilities to be secured.

Hardware Endpoints:

- Endpoint compromises today are frequently targeted at mobile devices, and much of the attention in the industry right now is focused on how to secure the mobile environment.
- The hardware phone or video codec, sitting quietly idle in the office but running 24/7, may, however, become an important tool for advanced corporate espionage, eavesdropping, or denial of service attacks.
- Modern VoIP phones have a fair bit of intelligence built into them and offer a previously unavailable avenue—some phones have a built-in layer two switch and are capable of executing XML scripts or Java code locally.
- Video codecs run all kinds of custom code required for video conferencing and content sharing and are sometimes directly exposed to the Internet.
- None of these devices have particularly robust mechanisms for authenticating to their control components unless a diligent administrator goes out of his or her way to enable them.
- Generally, these local capabilities are used to make the devices more interactive and functional, but they can be exploited in a variety of ways.
- According to the research firm Gartner, XML-based attacks are the next big thing, based on comments released after the disclosure of vulnerabilities related to remote code execution and DoS ability from exploited XML code.
- Part of what makes this a problem for the enterprise is the sheer number of endpoints connected to the system—a single phone system may manage thousands of endpoint devices, offering a massive exploitable base from which to wreak havoc via DDoS or other types of disruptive attacks.
- With VoIP in place, it not only disables your ability to make phone calls and causes productivity loss but also can compromise your entire enterprise network from within.

- Specialized endpoints are also employed for a variety of situations.
- Ensure that the vendors or OEMs supplying these components or devices have a suitable approach to security and understand their responsibility in the security of the overall infrastructure.
- It is important to recognize in this context that one phone can be the snowflake that starts the avalanche.

Software Endpoints:

- Enterprise desktop strategy focuses on convergence and extending simple and useful technologies to end users.
- This focus is intended to increase overall productivity and collaboration.
- One component of this strategy is the softphone or voice and video-enabled chat client.
- This is a piece of software that runs on a PC or mobile device and acts like a hardware endpoint by registering the call control element(s) as a device.
- Why would you install a soft client on a mobile device, which already has mobile capability? Two reasons: Cost is, of course, the first one.
- In many places, data usage on a cell phone is less costly than calling minutes, and by running a soft client, you convert what would otherwise be cellular usage minutes into an IP data stream (thank the "unlimited data plan" for this being a viable option).
- Second, by running the soft client, you can extend your enterprise features to the mobile user, including functionality not typically available on mobile devices such as consolidated extension-based or URI dialing.
- Some enterprises are even using direct inward system access (DISA) features or forking in order to make the mobile device itself an augmentation of the desk phone, creating a Single Number Reach (SNR) environment and automatically employing intelligent features like tail-end hop-off without direct user invocation.
- System administrators need to consider the fact that, although enabling these types of features is great for users and allows the unprecedented ability to control cost, the virtual voice security perimeter now extends well beyond the physical perimeter they are charged with managing, sometimes reaching around the globe and well outside of the traditional realms of control.
- Additionally, this trend mandates that much more granular attention be paid to the end-user computing environment.

Call and Contact Center Components:

- Call centers have made a remarkable evolutionary leap, from initially being used as a place to take orders and field complaints to, being a strategic asset that most enterprises cannot survive without.
- Within the last decade, call centers have morphed into "contact centers" and "centers of excellence."
- Trusted to sustain 24/7/forever operation and provide all levels of support to customers across every industry imaginable, these highly complex distributed systems, which now support millions of agents worldwide, have taken advantage of VoIP technologies in new and exciting ways—or, for the security administrator, in completely frightening ways.
- Their complexity has increased exponentially as the expectations of agents and customers alike have increased in sophistication.
- The two core components of any call center are automatic call detection (ACD) and interactive voice response (IVR).
- Simply put, the ACD moves calls around, and the IVR collects information from the caller and queues those calls in the appropriate places, based on defined variables such as agent skills.
- Whereas some systems simply queue calls and route them when an agent is available, others have advanced speech recognition capability and complicated algorithms predicting variables such as wait time for the next agent.
- Because of the complexity of these systems, it is especially important to ensure that they are patched and updated on a regular basis.
- A compromise of ACD or IVR could spell disaster for the victim, up to and including unrecoverable brand damage.
- Increasingly, these systems are being integrated with SaaS-based external solutions, especially CRM and other customer experience database systems.
- Although this offers the ability to drive a valuable and unique customer experience by having a single source of truth for customer data, it also warrants heavy scrutiny from a trained security professional.
- Many call centers employ predictive dialers or low-tech outbound dialers, which are powerful tools in the wrong hands unless best practices are followed to ensure that they are only allowed to call the numbers you want them to dial.
- Call recording and workflow management solutions can be very helpful for the overall productivity of your agent workforce, but they can also present a liability—these systems should have a known,

published policy for how they are used, how long data is stored, how archives are maintained, and what practices are used if data must or must not be destroyed.

Voicemail Systems:

- A major component of a VoIP-based telephony system is the voicemail system.
- Auto attendants, direct inward system access (DISA) features used for manual call forwarding, automatic call forwarding, and other voicemail features are a "standard" component of enterprise life, which nearly everyone has come to expect and rely on.
- Unfortunately, they have historically been one of the easiest systems to abuse for three main reasons:
    - Access to mailboxes is typically numeric-only, and people find long strings of numbers difficult to remember.
    - Easy (and often default) passwords are commonplace. War dialers can be set up to target these systems and record successful logins for attackers to return to later.
    - Anyone who has ever built a voicemail system knows the practice of initially setting everyone's default password to their extension, or perhaps the last four digits of their direct inward dialing (DID) phone number, or some other easy-to-figure-out formula.
    - This is a good opportunity to stretch your creative brain muscle and come up with something better.
    - Since voicemail systems have never really been considered a "key" component of enterprise infrastructure, much less attention has been paid to securing these systems than to, say, the enterprise ERP or financial systems.
    - Keep in mind, access to this type of functionality in the wrong hands can cause permanent damage to an organization in financial (and worse) ways.
    - More often than not system-level access to and from the outside world is not carefully controlled or audited, as some of a voicemail system's convenience "features" need outside access in order to work properly.
    - To preserve the sanctity of your voicemail system, always deactivate and preferably delete unused mailboxes, never leave default passwords in place, consider requiring more than a four-digit access code, and seriously evaluate how these systems will be used within your infrastructure.

9.1.3 VoIP Vulnerabilities and Countermeasures:

Having outlined the components that may fall under your purview in an enterprise VoIP infrastructure, let's now consider the three main exploitable paths from which you may be attacked:

- The "low-tech" hacks
- Attacks on server, appliance, or hardware infrastructure
- Advanced threats directed against specific systems or protocol
  - Telephony systems are frequently targeted partly because of the maturity of their services and partly owing to their sheer numbers.
  - Everyone has a phone system.
  - Here's what you can do to ensure that you've done your due diligence when it comes to protecting your VoIP and multimedia-rich infrastructure.
  - The following areas require specific attention from security administrators and these are the areas we'll focus on in this section:
- The original hacks—how to protect yourself from the oldest tricks in the book?
- Adding insult to injury: consider who tries to exploit voice services vs. VoIP services.
- Vulnerabilities and exploits:
  - The network
  - The servers
  - The appliances
  - The "other stuff"
- The protocols—examining specific areas of concern
- System integrators, hosted systems, and TEM as part of an enterprise security posture, Putting it all together: process makes perfect.

Old Dogs, Old Tricks: The Original Hacks:

- John Draper discovered (and exploited) a vulnerability in the Dual-tone multi-frequency signaling (DTMF) dialing systems of the time when he found that a toy whistle from a cereal box could be used to produce a 2600 Hz sound to manipulate the communication protocol of public phone systems to obtain free long distance.
- He was sentenced to two months in prison.
- You might think that telephone companies would have immediately

fixed the vulnerability so other people couldn't repeat the exploit.

- Low-tech approaches like this worked for many phone systems around the globe well into the 1980s.
- While most modern IP-based systems are smart enough not to fall for the old DTMF tricks, you want to take precautions against equally simple attacks that will probe your defenses on a daily basis.
- Information on exploits of various systems is so readily available, that taking advantage of open relays is a common recreational and for-profit activity.
- In addition, the security of a fixed location, such as a landline, is no longer a reliable way to ensure that you know where a call is originating from, an important part of understanding what someone is trying to do.
- The portability of public IP address space means that spoofing the physical location of a phone is a relatively easy matter, and tracking it down can be quite difficult.
- The VoIP predator's basic approach is to sell a VoIP service to end customers and then use compromised systems to route those calls for free from and to virtually anywhere.
- The predator charges for service on the front end but gets a free service on the back end.
- There's always a phone bill—but it is generally left up to the victim to settle, as the victim's carrier has to pay their partner provider for the calls regardless.
- In the enterprise, the trick is to not become one of those relays. Often, people or businesses think they are subscribing to a legitimate service, as there are hundreds, even thousands, of exploited gateways.
- Of little help is the fact that hiding voice transit and routing among other IP-based traffic is easy.

Assessment Audit:

Create a risk profile for low-tech hacks in your organization by doing the following audit.

- What is your external facing profile?
- Are there exposed numbers that can reach internal systems and access them?
- If so, do those internal systems have password or PIN protection? What complexity?
- If simple access is required for any reason, can you audit access?

- Who is responsible for accepting the risk of a breach?
- Is this person aware of this responsibility and what it means to the organization?
- Have you performed an inventory of all voice protocols enabled on your gateways for use later? If not, do so now.
- Is DISA enabled?
- Given that some organizations prefer a "live answer" experience for their internal and external customers, have the operators been trained and given process documentation to follow in the event of a suspected malicious call?
- War dialers are still out there … do you have the capability to determine if someone is trying to breach your defenses? Then use it.
- Enlist the phone company's help in tracking down malicious behavior before the culprit finds an opening.
- Do you have a Telecom Expense Management (TEM) program that tracks and reports on the costs of phone usage and identifies which phones have the largest bills?

Action Steps:

1. Create a scorecard from the information you've gathered from your audit in order to identify your most significant risks and areas in need of attention; prioritize high-risk items with a standard likelihood and severity graph or matrix.
2. Know your dial-in numbers; only publish them for those who may need to use them, and ensure the executive team is aware of the risk of offering this service.
3. Enforce password requirements for system access.
4. Delete old and unused mailboxes as soon as possible.
5. Use restrictions to prevent DISA from being used for long-distance and international calling; if not possible or if the feature is needed, ensure that all calls made via DISA are logged and auditable and users with access to the service are educated on the risks.
6. Limit exposure where possible by using fewer external dial-in numbers; enforce a business process that requires security team review and approval prior to enabling new services.
7. Do not offer all user features to all users by default, unless your security program can support the ongoing use, auditing, and management of these features for the full user population.
8. Pay attention to call forwarding and who is allowed to use the feature to send calls outside of your perimeter.

9. Determine how your TEM program can flag abnormal patterns or utilization in order to give you visibility into when you may have a problem.

Vulnerabilities and Exploits:

For our purposes in this section, vulnerability means a weakness that has not yet been used to compromise a perimeter, whereas exploit is a compromised vulnerability.

Network:

- Security administrators need to understand how to strike a balance between functionality and security, particularly when their peers (network and systems administrators) have the job of trying to move traffic in an unobstructed fashion across common multiaccess networks.
- Inspecting packets takes resources and adds transit time, which can lead to an adversarial relationship between the teams working to move packets from place to place seamlessly and the teams trying to ensure that legitimate data is contained within those packets.
- Sit down with the parties responsible for the network and the voice systems, and use a cooperative approach citing the greater good.

Discuss the following topics:

- What protocols will be allowed and used for VoIP on the network?
- What protocols should be explicitly blocked?
- How much bandwidth is "normal" for your call volumes?
- If you're using a G.711 codec, you should expect ~80 kb per call.
- G.729 can vary depending on the compression used and specific subprotocol.
- Can you create segregated security areas (zones) for your voice components?
- Subnets for voice control and voice gateways.
- Subnets for phones (many network switches now have a voice VLAN command that allows the phone to exist on a different VLAN than the device attached behind the phone).
- Only allow the protocols in and out that you need; if a system integrator (SI) is implementing the system for you, have them provide this information, or consult your system documentation from the manufacturer.

Servers:

- As with any server-based system, understand your key weaknesses and most vulnerable areas.
- As described in the previous section, having updated diagrams and an inventory of all of the components of your voice platform will help ensure that those assets can be secured in a reasonable way.
- Documentation is a critical but frequently overlooked part of a security management strategy, which applies to VoIP as well.
- For any server-based system that runs on a commodity OS (typically Windows or Unix), ensure that your network or server teams are prepared to follow patch management procedures for these resources along with the rest of the environment.
- With companies like Microsoft enabling features like enterprise voice services and voicemail, system administrators have the added responsibility of ensuring that Windows servers are patched for these in addition to the rest of their KB patches.
- In addition, many contact center and workforce productivity solutions, some of which have special versions supported only by the manufacturer, also run-on Windows under the hood.

Appliances:

- Once upon a time, when DTMF ruled the voice world, dialogic boards were the key for interpreting dialed digits, and every voice system had them in either the PBX controller or voicemail system.
- Back then, nearly all voice systems would have been considered "custom appliances" by today's standards.
- The common modern practice for many manufacturers today is to buy OEM hardware from one of the big server suppliers and to run either a proprietary OS or a custom version of a commodity operating system to create their "appliance."
- Some voice hardware providers still make their own application-specific integrated circuit (ASIC) chips and hardware chassis, but this is becoming less common as standardization and virtualization gain adoption in the voice space.
- The real relevance for security administrators is in the amount of customization the provider does in order to offer their features.
- In one sense, a certain "security by obscurity" is achieved with highly customized platforms because there are generally fewer of them in the field and they present a less attractive target than something more widely deployed.

- Inversely, an exploit specific to a unique platform may remain undiscovered for a longer period of time, as you are dependent on the manufacturer or specific product community to identify such vulnerabilities.

Everything Else:

- There's a lot that falls into the "other stuff" category, from hosted systems to all of the components that are not considered call control.
- Hosted systems are covered later, as they require special considerations.
- The two most commonly exploited systems in the "other stuff" category are DISA-enabled voicemail servers and gateways that allow connections from the Internet.
- No matter what brand of phone system you are running, keep the following information handy:

For the voicemail system:

- Use a least-privilege model in which administrators do not have mailboxes accessible via external means; require a VPN and strong authentication.
- Delete unused mailboxes.
- Force complexity requirements for voicemail passwords and access codes
- Carefully consider the risks of allowing remote call forwarding or other call forwarding features, particularly those that can be enabled remotely; if a feature is not absolutely necessary for your users, do not allow it.
- Use strong authentication for "remote destination" calling or calling-card-type features.

For the voice gateways:

- Explicitly disable unused services, especially those with Internet-facing connections.
- Lockdown via ACL or firewall what systems are allowed to communicate with the gateways via IP; use a secondary system (IPS) to watch what the gateways are doing if you are running SIP or a similar protocol.

The Protocols:

- At the heart of the family of VoIP, technologies are the specific protocols that enable the transit and real-time conversations that IP networks were not originally designed to handle.

- Security filtering and analysis for most network-based communications have become quite advanced, but VoIP-specific capability has not kept up with the rest of the industry.
- While current-generation firewall ALGs can tell you that a VoIP conversation is, in fact, a valid protocol (RTP, RTSP) and they cannot:
    - Tell you what is taking place in that conversation
    - Guarantee that no one else is listening in
    - Determine that a voice conversation is the only thing taking place over that communication channel
- Outside of the U.S. Department of Defense or Department of Homeland Security, advanced heuristic electronic listening is not widely employed for security purposes.
- Realistically and within the reach of ordinary organizations, the following section lists the mechanics of the protocols you'll encounter on an enterprise network, some associated risks, and practical suggestions for protecting them.

Protocol: SIP:

Governing Standard:

- The Session Initiation Protocol (SIP) standards and extensions are so numerous that an RFC is dedicated to identifying all of the other SIP RFCs, and there are books to help navigate the situation.
- For the basics, RFC 3261 is the core SIP standard.
- SIP is a highly complex set of protocols—really a protocol suite with volumes dedicated to implementing, managing, and securing the entire stack based on different use cases**.**
- This overview is not a substitute for deeper research on how SIP is being used within an enterprise and the methods required to ensure it has been securely implemented and suitably protected.

Purpose:

- Application layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants.
- Sessions include Internet telephone calls, multimedia calls and distribution, and multimedia conferencing.
- In plain English: SIP is used for all kinds of voice and multimedia applications and is prolific both on corporate networks and the Internet, sometimes appearing unintentionally in enterprise environments via voice-enabled chat clients that are both sponsored (e.g., Lync, Connect, Jabber, etc.) and unauthorized (Yahoo messenger, AIM, etc.).

Function:

- SIP is a session-based protocol that uses SIP invitations that are used to create sessions.
- These carry session descriptions that allow participants to negotiate a set of compatible media types.
- SIP makes use of proxy servers to route requests to a user's registered location, authenticate and authorize services, implement provider call-routing policies, and provide features.
- SIP also provides a registration function that allows users to upload their current locations for use by SIP proxies.
- SIP runs on top of several different transport protocols and relies on a variety of different mechanisms for security**.**

Known Compromises and Vulnerabilities:

Because there are so many SIP-related vulnerabilities that exist based on the different implementations of the protocol and extensions, it is worth classifying them into the following categories:

- Control system and SIP proxy
- Device-based (including a mobile device)
- DoS, DDoS, flooding
- SPAM over Internet Telephony (SPIT)
- Vishing (the criminal practice of using social engineering over a telephony system, widely facilitated by VoIP and SIP-based systems)
- Spoofing, barging, and redirection
- Replay and interception

Recommendations:

- If you're going to allow SIP on the network or enable SIP-based enterprise applications, either for voice and video or other converged services or for less specific uses third-party IM clients, etc., seriously consider the minimum level users need in order to function.
- Discuss this with whoever in your organization is responsible for the services that use SIP and ensure that they understand the risks of this highly dynamic protocol.
- If SIP is required, and particularly if such a requirement includes SIP services be available via the Internet, ensure you are using a device that has the capability to inspect the traffic and validate that the information in the SIP header is correctly formed and is accurate.

- This is the easiest way to tell if there is a spoof attempt or other malicious activity in process.

9.1.4 PBX:

A Private Branch Exchange (PBX) is a computer-based switch that can be thought of as a local phone company.

Following are some common PBX features:

- Multiple extensions
- Voicemail
- Call forwarding
- Fax management
- Remote control (for support)

Hacking a PBX:

Attackers hack PBXs for several reasons:

- To gain confidential information (espionage)
- To place outgoing calls that are charged to the organization's account (and thus free to the attacker)
- To cause damages by crashing the PBX

Administrative Ports and Remote Access:

- Administrative ports are needed to control and diagnose the PBX.
- In addition, vendors often require remote access via a modem to be able to support and upgrade the PBX.
- This port is the number one hacker entry point.
- An attacker can connect to the PBX via the modem; or if the

administrative port is shared with a voice port, the attacker can access the port from outside the PBX by calling and manipulating the PBX to reach the administrative port.

Voicemail:

- An attacker can gain information from voicemail or even make long-distance phone calls using a "through-dial" service.
- After a user has been authenticated by the PBX, that user is allowed to make calls to numbers outside the PBX.
- An attacker can discover a voicemail password by running an automated process that "guesses" easy passwords such as "1111," "1234," and so on.

Denial of Service:

A PBX can be brought down in a few ways:

- PBXs store their voicemail data on a hard drive.
- An attacker can leave a long message, full of random noises, in order to make compression less effective—whereby a PBX might have to store more data than it anticipated.
- This can result in a crash.
- An attacker can embed codes inside a message.

Securing a PBX:

Here is a checklist for securing a PBX:

- Connect administrative ports only when necessary.
- Protect remote access with a third-party device or a dial-back.
- Review the password strength of your users' passwords.
- Allow passwords to be different lengths, and require the # symbol to indicate the end of a password, rather than revealing the length of the password.
- Disable all through-dialing features.
- If you require dial-through, limit it to a set of predefined needed numbers.
- Block all international calls, or limit the number of users who can initiate them.
- Block international calls to places such as the Caribbean that fraudsters tend to call.

- Train your help desk staff to identify attempted PBX hacks, such as excessive hang-ups, wrong number calls, and locked-out mailboxes.
- Make sure your PBX model is immune to common DoS attacks.

9.1.5 TEM: Telecom Expense Management:

- Telecom Expense Management is both a methodology and a software platform that permits an organization to manage expenses tied to the critical strategic assets of its telecom network.
- It encompasses the technology, processes, policy, and people required to manage and use a business telecommunications system.

The technology and services involved include:

- Mobile services, Pagers
- Voice lines, PRI, VoIP
- Data Circuits, DSL, T1, T3, MPLS
- Calling cards
- Conferencing (Audio, Video, and Web)

TEM is offered as software as a service (SaaS) where vendors are able to target net savings for clients and supports the following practices:

- Invoice and auditing & processing
- Procurement (Provisioning and escalation)
- Inventory validation and asset management
- Mobile usage policy (Implementation and management)
- Integration of mobile and landline systems

- Management of multi-currency, multi-vendor system
- Contract management and compliance
- Reporting with business intelligence and role-based flow

The goals of TEM are to streamline management, validate assets, and provide data on costs and billing for efficient management of accountability, mobile services, and cost savings.

- Phone bills can be more complex to read than ancient hieroglyphs, and there has been little progress made in simplifying or decoding them for the average consumer or telecom manager.
- Understanding what is on your phone bill so you can tell whether your voice providers are doing the right thing is important (there are alarming statistics on the error percentage in consumer and corporate phone bills).
- But that's the job of your telecom group—why would a security professional care about phone bills? Your phone bill can have some clues to other problems in your environment, and a TEM program can help automate the process of getting to the goodies, the high-quality information you need to tell quickly if you have a security problem related to your phone system.
- TEM is a relatively new discipline in the telephony space, gaining major adoption within the last decade.
- There are many firms armed with specialized software ready to help you collect, organize, understand, interpret, and audit your telephone bills, all for a modest gain-share or percentage of savings fee.
- While effort is involved in the setup and optimization of the billing, once you've reached the point where a TEM firm can actually audit bills, you're likely to have a useful tool to spot irregular or suspicious activity that may otherwise be tough to catch.
- At some point in his or her career, every security professional gets pulled into a conversation about some malicious phone calls or fraudulent billing.
- Even if the administrator hasn't had much to do with telecom prior to that, suddenly he or she has to figure out how the telephone fraud happened.
- With TEM in place, the security administrator has a powerful tool to search for precursors or other suspicious activity that could be related to the exploited vector the attacker used and can help identify where it may happen again.
- If e.g., an unexpected Rs.10,000 phone bill arrives out of nowhere with calls to countries your users have no reason to call, and through

investigation, you determine that it was the result of a gateway compromise, you could use the TEM capability to check the rest of the PRI or voice services globally to determine if any of the same suspicious or exploited numbers were being called and to help determine if there are other potentially compromised gateways.

- You would, of course, also want to do an internal network audit of the services and security on the gateways themselves, as you'll want to plug the holes you know about at the same time that the TEM and audit function is checking for leaks elsewhere for you.
- Although phone bills are generally not directly related to the security group's main role, it is the objective of every security group to protect stakeholder interests, and TEM can help a security group detect anomalous behavior and operate more quickly and effectively when they are called in to action for this type of an issue.

9.2 LETS US SUM UP

- Successful security administrators should keep this mantra in mind in all things that they do: process, process, process.
- Having solid, repeatable processes to support any efforts on which they embark can not only help to build trust in the security group but also help elevate the level to which security supports and enables the business.
- Specifically with voice systems, investing the time to create a process cycle for evaluating new voice initiatives and maintaining updated documentation will pay dividends in the long run.
- Voice systems warrant special attention from security groups.

9.3 QUESTIONS

1. Explain the main functionality of VoIP.
2. Explain the different components of VoIP.
3. Explain the features of different protocols of VoIP.
4. What is the role and functionality of PBX?
5. Write a note on TEM.
6. Explain SIP in details.

9.4 BIBLIOGRAPHY

- Dwivedi, Himanshu. Hacking VoIP: Protocols, Attacks, and Countermeasures. No Starch Press, 2008

- The Complete Reference: Information Security – The Complete Reference: Information Security, McGraw Hill

9.5 REFERENCE

- https://www.researchgate.net/publication/313101329_The_Evaluation_of_Voice-over_Internet_Protocol_VoIP_by_means_of_Trixbox
- https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1705876

*****

10

OPERATING SYSTEM SECURITY MODELS

Unit Structure

10.0 OBJECTIVES

After going through this unit, you will be able to:

- Understand the security reference monitor and how it manages the security of its related elements
- Aware of access control—the heart of information security
- Learn International standards for operating system security, which provide organizations with a level of assurance and integrity.

10.1 INTRODUCTION

An operating system security model is the foundation of the operating system's security functionality. All security functionality is architected, specified, and detailed in advance—before a single line of code is written. Everything built on top of the security model must be mapped back to it, and any action that violates the security model should be denied and logged.

Protection and security require that computer resources such as CPU, software, memory, etc. are protected. This extends to the operating system as well as the data in the system. This can be done by ensuring integrity, confidentiality, and availability in the operating system.

10.1.1 Operating System Models:

- The operating system security model also known as the trusted computing base, or TCB is simply the set of rules, or protocols, for

security functionality.

- Security commences at the network protocol level and maps all the way up to the operations of the operating system.
- An effective security model protects the entire host and all of the software and hardware that operate off it.
- Previous systems used an older, monolithic design, which proved to be less than effective.
- Current operating systems are optimized for security, using a compartmentalized approach.
- The trend in operating systems has been toward a microkernel architecture.
- In contrast to the monolithic kernel, microkernels are platform-independent.
- Although they lack the performance of monolithic systems, they are catching up in terms of speed and optimization.
- A microkernel approach is built around a small kernel with a common hardware level.
- The key advantage of a microkernel is that the kernel is small and easy to port to other systems.

The Underlying Protocols Are Insecure:

- Extending the submarine analogy, the security protocol has a direct connection to the communication protocol.
- Today, the protocol is TCP/IP—the language of the Internet and, clearly, the most popular and utilized protocol.
- If the operating system is an island, then TCP/IP is the sea.
- Given that fact, any operating system used today must make up for TCP/IP's shortcomings.
- Even the best operating system security model can't operate in a vacuum or as an island, however.
- If the underlying protocols are insecure, then the operating system is at risk.
- What's frightening about this insecurity is that while the language of the Internet is TCP/IP, effective security functionality was not added to TCP/IP until version 6 in the late 1990s.
- Given that the vast majority of the Internet is still running an insecure version of TCP/IP, version 4, the entire Internet and corporate

computing infrastructure is built on and running on an insecure infrastructure and foundation.

- The TCP/IP protocol's main problems are as follows:

Vulnerable to spoofing:

- Spoofing is the term for establishing a connection with a forged sender address.
- Normally this involves exploiting trust relations between the source address and the destination address.
- The ability to spoof the source IP address assists those carrying out DoS attacks by making it difficult for victims to block the DoS traffic, and the predictability of the initial sequence number (ISN), which is a unique number that is supposed to guarantee the authenticity of the sender, contributes more to spoofing attacks by allowing an attacker to impersonate legitimate systems and take over a connection (as in a man in the middle attack).

Vulnerable to session hijacking:

- An attacker can take control of a connection by intercepting the session key and using it to insert his own traffic into an established TCP/IP communication session, usually in combination with a DoS attack against the legitimate sender so that traffic cannot get through, as in a man in the middle attack.

Predictable sequence guessing:

- The sequence number used in TCP connections is a 32-bit number, so the odds of guessing the correct ISN would seem to be exceedingly low.
- If the ISN for a connection is assigned in a predictable way, however, it becomes relatively easy to guess.
- The truth is that the ISN problem is not a protocol problem but rather an implementation problem.
- The protocol actually specifies pseudorandom sequence numbers, but many implementations have ignored this recommendation.

No authentication or encryption:

- The lack of authentication and encryption with TCP/IP is a major weakness.

Vulnerable to SYN flooding:

- SYN flooding takes advantage of the three-way handshake in establishing a connection.
- When Host B receives an SYN request from A, it must keep track of

the partially opened connection in a listen queue, enabling successful connections even with long network delays.

- The problem is that many implementations can keep track of only a limited number of connections.
- A malicious host can exploit the small size of the listen queue by sending multiple SYN requests to a host but never replying to the SYN and ACK sent by the other hosts.
- By doing so, the malicious host quickly fills up the other host's listen queue, and that host stops accepting new connections until a partially opened connection in the queue is completed or times out.

The security benefits of TCP/IP version 6 include:

- IP Sec security
- Authentication and encryption
- Resilience against spoofing
- Data integrity safeguards
- Confidentiality and privacy

An effective security model recognizes and is built around the fact that because security is such an important design goal for the operating system, every resource that the operating system interfaces with memory, files, hardware, device drivers, and so on must interact from a security perspective. By giving each of these objects an access control list (ACL), the operating system can detail what that object can and can't do by limiting its privileges.

Access Control Lists:

- Much of the security functionality afforded by an operating system is via the ACL.
- Access control comes in many forms, but in whatever form it is implemented, it is the foundation of any security functionality.
- Access control enables you to protect a server or parts of the server such as directories, files, file types, and so on.
- When the server receives a request, it determines access by consulting a hierarchy of rules in the ACL.
- An access control list is defined as a table that tells a computer operating system which access rights each user has for a particular system object, such as a file directory or an individual file.
- Each object has a security attribute that identifies its access control list.

- The list has an entry for each system user with access privileges.
- The most common privileges include the ability to read a file or all the files in a directory, to write to the file or files, and to execute the file if it is an executable file or program.
- Each operating system implements the ACL differently.
- In Windows, an ACL is associated with each system object.
- Each ACL has one or more access control entries (ACEs), each consisting of the name of a user or a group of users.
- The user can also be a role name, such as a programmer or tester.
- For each of these users, groups, or roles, the access privileges are stated in a string of bits called an access mask.
- Generally, the system administrator or the object owner creates the access control list for an object.
- Each ACE identifies a security principal and specifies a set of access rights that are allowed, denied, or audited for that security principal.

An object's security descriptor can contain two ACLs:

- A discretionary access control list (DACL) that identifies the users and groups who are allowed or denied access
- A system access control list (SACL) controls how access is audited. Unix systems also have access control based on user permissions and roles defined by groups.
- System objects have permissions defined within them, which can be controlled on the basis of read, write, and execute permissions for each user or group defined on the system.

MAC vs. DAC:

- Access control lists can be further refined into both required and optional settings.
- This refinement is carried out more precisely with discretionary access control and is implemented by discretionary access control lists (DACLs).
- The difference between discretionary access control and its counterpart, mandatory access control, is that DAC provides an entity or object with access privileges it can pass to other entities.
- Depending on the context in which they are used, these controls are also called rule-based access control (RBAC) and identity-based access control (IBAC).
- Mandatory access control requires that access control policy decisions be beyond the control of the individual owners of an object.

- MAC is generally used in systems that require a very high level of security.
- With MAC, only the administrator and not the owner of the resource may make decisions that bear on or derive from the security policy.
- Only a security administrator may change a resource's category, and no one may grant a right of access that is explicitly forbidden in the access control policy.
- MAC is always prohibitive and not permissive.
- Only within that context do discretionary controls operate, prohibiting still more access with the same exclusionary principle.
- All of the major operating systems such as Solaris, Windows, NetWare, and so on use DAC.
- MAC is implemented in more secure, trusted operating systems such as Trusted BSD and Trusted Solaris.

Table Below details the difference in functionality between discretionary and mandatory access control:

Control Type	Functionality
Discretionary	• Individual users may determine the access controls. • Works well in the commercial and academic sectors. • Not suited for the military. • Effective for private websites, etc.
Mandatory	• Allows the system administrator to set up policies and accounts that will allow each user to have full access to the files and resources needed, but no access to other information and resources • Not immediately necessary to perform assigned tasks. • Site-wide security policy is enforced by the system in addition to the discretionary access controls. • Better suited to environments with rigid information. • Effective access restrictions. • Access permission cannot be passed from one user to another. • Requires labeling: sensitivity and integrity labels.

Table: The Difference in Functionality Between Discretionary and Mandatory Access Control

10.1.2 Classic Security Models:

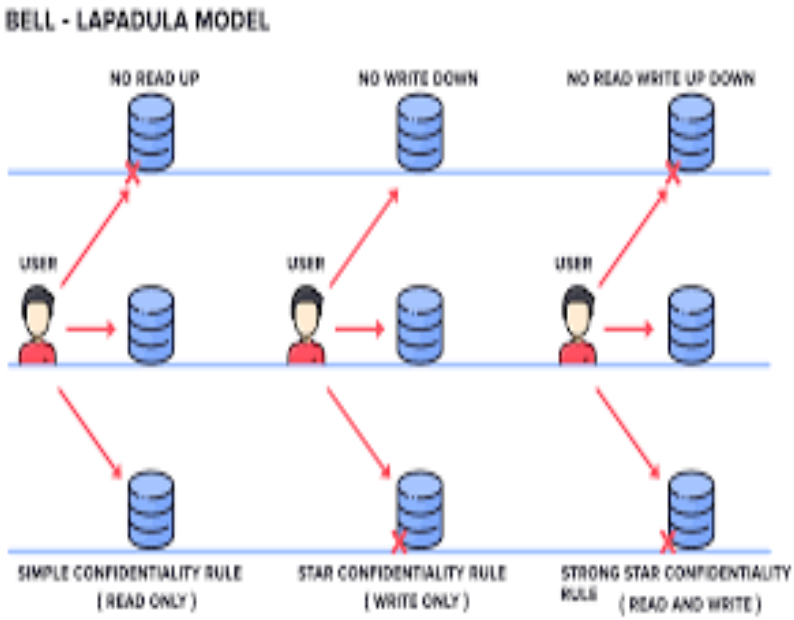
The following three most famous security models in computer security are:

- Bell-LaPadula
- Biba
- Clark-Wilson

Those designing operating system security models have the liberty of picking and choosing from the best of what the famous models have, without being encumbered by their myriad details.

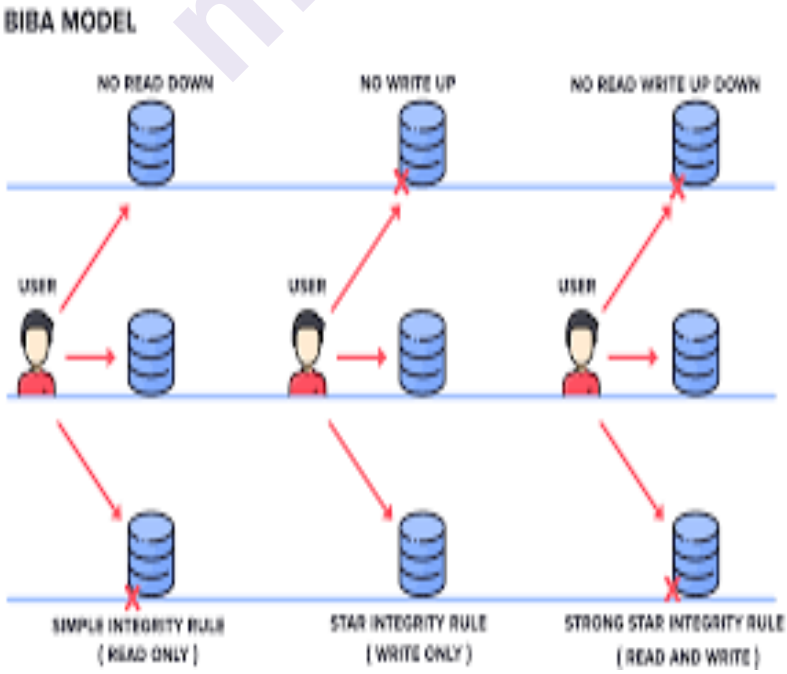
Bell-LaPadula:

- While the Bell-LaPadula model was revolutionary when it was published in 1976, descriptions of its functionality today are almost anticlimactic.
- The Bell-LaPadula model was one of the first attempts to formalize an information security model.
- The Bell-LaPadula model was designed to prevent users and processes from reading above their security level.
- This is used within a data classification system—so a given classification cannot read data associated with a higher classification—as it focuses on the sensitivity of data according to classification levels.
- In addition, this model prevents objects and processes with any given classification from writing data associated with a lower classification.
- This aspect of the model caused a lot of consternation in the security space.
- Most operating systems assumed that the need to write below one's classification level is a necessary function.
- But the military influence on which Bell-LaPadula was created mandated that this be taken into consideration.



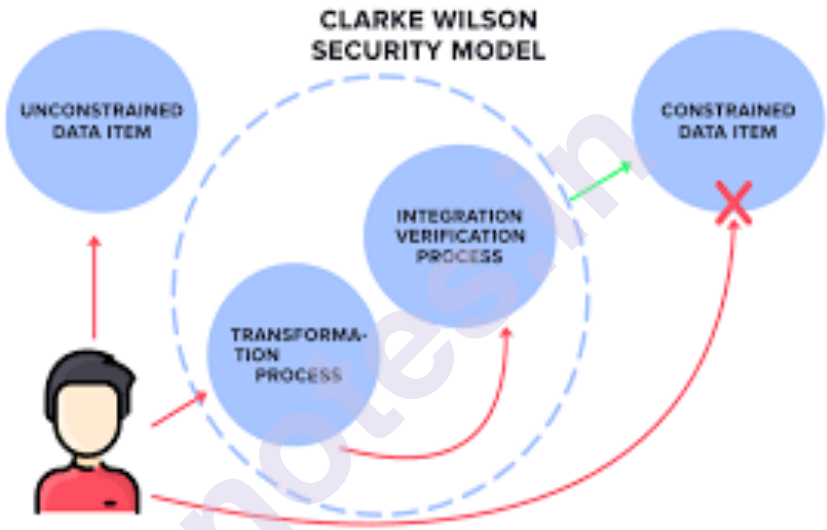
Biba:

- Biba is often known as a reversed version of Bell-LaPadula, as it focuses on integrity labels, rather than sensitivity and data classification.
- Bell-LaPadula was designed to keep secrets, not to protect data integrity.
- Biba covers integrity levels, which are analogous to sensitivity levels in Bell-LaPadula, and the integrity levels cover inappropriate modification of data.
- Biba attempts to preserve the first goal of integrity, namely to prevent unauthorized users from modifying data.



Clark-Wilson:

- Clark-Wilson attempts to define a security model based on accepted business practices for transaction processing.
- Much more real-world-oriented than the other models described, it articulates the concept of well-formed transactions that:
  - Perform steps in order
  - Perform exactly the steps listed
  - Authenticate the individuals who perform the steps



TCSEC:

- In the early 1970s, the United States Department of Defense published a series of documents to classify the security of operating systems, known officially as the Trusted Systems Security Evaluation Criteria.
- The TCSEC was heavily influenced by Bell-LaPadula and classified systems at levels A through D.
- TCSEC was developed to meet three objectives:
- To give users a yardstick for assessing how much they can trust computer systems for the secure processing of classified or other sensitive information
- To guide manufacturers in what to build into their new, widely available commercial products to satisfy trust requirements for sensitive applications
- To provide a basis for specifying security requirements for software and hardware acquisitions.

Although TCSEC offered a lot of functionality, it was, by and large, not suitable for the era of client/server computing. The client/server computing world was embryonic when the TCSEC was created, although its objectives were admirable.

The table below provides a brief overview of the different classification levels:

TCSEC Rating	Usage
D—Minimal Protection	• Any system that does not comply with any other category or has failed to receive a higher classification • No security requirements • Was used as a catch-all category for such operating systems as MS-DOS and Windows 95/98/ME
C1—Discretionary Protection	• DACL/ACL • User/Group/World Protection • Usually for users who are all on the same security level • Protected operating system and system operations mode • Periodic integrity checking of TCB • Tested security mechanisms with no obvious bypasses • Documentation for user security • Documentation for systems administration security • Documentation for security testing • TCB design documentation
C2—Controlled Access Protection	**Everything in C1 plus:** • Object protection can be on a single-user basis, for example, through an ACL or trustee database • Authorization for access may be assigned only by authorized users • Object reuse protection • Mandatory identification and authorization procedures for users, such as username/password • Full auditing of security events • Protected system mode of operation • Added protection for authorization and audit data • Documentation as C1 plus information on examining audit information • One of the most common certifications, including VMS, IBM OS/400, Windows NT 3.51, Novell NetWare 4.11, Oracle 7, and DG AOS/VS II

B1—Labeled Security Protection	**Everything in C2 plus:** • Mandatory security and access labeling of all objects, for example, files, processes, devices • Label integrity checking (for example, maintenance of sensitivity labels when data is exported) • Auditing of labeled objects • Mandatory access control for all operations • Enhanced auditing • Enhanced protection of operating systems • Improved documentation • Operating systems: HP-UX BLS, Cray Research Trusted Unicos 8.0, Digital SEVMS, Harris CS/SX, and SGI Trusted IRIX

Table: Classifications of Operating Systems Security

TCSEC Rating	Usage
B2—Structured Protection	**Everything in B1 plus:** • Notification of security level changes affecting interactive users • Hierarchical device labels • Mandatory access over all objects and devices • Trusted path communications between user and system • Tracking down of covert storage channels • Tighter system operations mode into multilevel independent units • Covert channel analysis • Improved security testing • Formal models of TCB • Version, update, and patch analysis and auditing • Example systems: Honeywell Multics and Trusted XENIX
B3—Security Domains	**Everything in B2 plus:** • ACL additionally based on groups and identifiers • Trusted path access and authentication • Automatic security analysis • TCB models more formal • Auditing of security auditing events • Trusted recovery after the system was down and relevant documentation

	• Zero design flaws in TCB and minimum implementation flaws • Only B3-certified OS is Getronics/Wang Federal XTS-300
A1—Verified Design	A1 is the highest level of certification and demands a formal security verification method to ensure that security controls protect classified and other sensitive information. At this level, even the National Security Agency cannot break in. **A1 requires everything in B3 plus:** • Formal methods and proof of the integrity of TCB Label integrity checking (for example, maintenance of sensitivity labels when data is exported) • Only A1-certified systems: Gemini Trusted Network Processor and Honeywell SCOMP

Table: Classifications of Operating Systems Security (continued)

Labels:

- TCSEC makes heavy use of the concept of labels.
- Labels are simply security-related information that has been associated with objects such as files, processes, or devices.
- The ability to associate security labels with system objects is also under security control.
- Sensitivity labels, used to define the level of data classification, are composed of a sensitivity level and possibly some number of sensitivity categories.
- The number of sensitivity levels available is dependent on the specific operating system.
- In a commercial environment, the label attribute could be used to classify, for example, levels of a management hierarchy.
- Each file or program has one hierarchical sensitivity level. A user may be allowed to use several different levels, but only one level may be used at any given time.
- While sensitivity labels identify whether a user is cleared to view certain information, integrity labels identify whether data is reliable enough for a specific user to see.
- An integrity label is composed of an integrity grade and some number of integrity divisions.
- The number of hierarchical grades to classify the reliability of information is dependent on the operating system.

- While TCSEC requires the use of labels, other regulations and standards such as the Common Criteria also require security labels.
- There are many other models around, including the Chinese wall, Take-Grant, and more.
- But in practice, none of these models has found favor in contemporary operating systems (Linux, Unix, Windows)—they are overly restrictive and reflect the fact that they were designed before the era of client/server computing.
- Current operating system architects are able to use these references as models, pick and choose the best they have to offer, and design their systems accordingly.

10.1.3 Reference Monitor:

- The Computer Security Technology Planning Study Panel called together by the United States Air Force developed the reference monitor concept in 1972.
- They were brought together to combat growing security problems in a shared computer environment.
- In 1972, they were unable to come up with a fail-safe solution; however, they were responsible for reshaping the direction of information security today.

The Reference Monitor Concept:

The National Institute of Standards and Technologies describes the reference monitor concept as an object that maintains the access control policy.

It does not actually change the access control information; it only provides information about the policy. The security reference monitor is a separable module that enforces access control decisions and security processes for the operating system.

All security operations are routed through the reference monitor, which decides if the specific operation should be permitted or denied. Perhaps the main benefit of a reference model is that it can provide an abstract model of the required properties that the security system and its access control capabilities must enforce.

The main elements of an effective reference monitor are that it is:

- **Always on**: Security must be implemented consistently and at all times for the entire system and for every file and object.
- **Not subject to preemption**: Nothing should be able to preempt the reference monitor. If this were not the case, then it would be possible for an entity to bypass the mechanism and violate the policy that must be enforced.

- **Tamperproof:** It must be impossible for an attacker to attack the access mediation mechanism such that the required access checks are not performed and authorizations are not enforced.
- **Lightweight:** It must be small enough to be subject to analysis and tests, proving its effectiveness.

The reference monitor concept has proved itself to be a useful tool for computer security practitioners. It can also be used as a conceptual tool in computer security education.

Windows Security Reference Monitor:

- The Windows Security Reference Monitor (SRM) is responsible for validating Windows process access permissions against the security descriptor for a given object.
- The Object Manager then, in turn, uses the services of the SRM while validating the process's request to access any object.
- Windows is clearly not a bulletproof operating system, as is evident from the number of security advisories alone.
- In fact, it is full of security holes.
- But the fact that it is the most popular operating system in use in corporate settings and that Microsoft has been, for the most part, open with its security functionality, makes it a good case study for a real-world example of how an operating system security model should operate**.**

10.1.4 Trustworthy Computing:

The four goals of the Trustworthy Computing initiative are:

Security:

As a customer, you can expect to withstand attack. In addition, you can expect the data is protected to prevent availability problems and corruption.

Privacy:

You have the ability to control information about yourself and maintain the privacy of data sent across the network.

Reliability:

When you need your system or data, they are available.

Business integrity:

The vendor of a product acts in a timely and responsible manner, releasing security updates when a vulnerability is found.

To track and assure its progress in complying with the Trustworthy Computing initiative, Microsoft created a framework to explain its objectives:

1. Its products are secure by design,
2. Secure by default, and
3. Secure in deployment and that it provides communications.

Secure by design simply means that all vulnerabilities are resolved prior to shipping the product.

Secure by design requires three steps:

1. Build a secure architecture. This is imperative. Software needs to be designed with security in mind first and then features.
2. Add security features. Feature sets need to be added to deal with new security vulnerabilities.
3. Reduce the number of vulnerabilities in new and existing code. The internal process at Microsoft was revamped to make developers more conscious of security issues while designing and developing software.

Secure deployment means ongoing protection, detection, defense, recovery, and maintenance through good tools and guidance. Communication is the key to the whole project.

10.1.5 International Standards for Operating System Security:

- Although Microsoft's Trustworthy Computing initiative has been heralded as a giant step forward for computer security, much of the momentum started years earlier.
- And one of the prime forces has been the Common Criteria. The need for a common information security standard is obvious.
- Security means many things to different people and organizations. But this subjective level of security cannot be objectively valuable.
- Therefore, common criteria were needed to evaluate the security of an information technology product.

Common Criteria:

- The need for a common agreement is clear.
- When you buy a DVD, put gas in your car, or make a purchase from an online retailer, all of these activities function because they operate in accordance with a common set of standards and guidelines.
- And that is precisely what the Common Criteria are meant to be, a global security standard ensuring that there is a common mechanism for evaluating the security of technology products and systems.

- By providing a common set of requirements for comparing the security functions of software and hardware products, the Common Criteria enable users to have an objective yardstick by which to evaluate a product's security.
- Common Criteria certification is slowly but increasingly being used as a touchstone for many Requests for Proposals, primarily in the government sector.
- By offering a consistent, rigorous, and independently verifiable set of evaluation requirements for hardware and software, Common Criteria certification is intended to be the Good Housekeeping seal of approval for the information security sector.
- But what is especially historic about the Common Criteria is that this is the first-time governments around the world have united in support of an information security evaluation program.

Common Criteria Origins:

- In the United States, the Common Criteria have their roots in the Trusted Computer System Evaluation Criteria (TCSEC), also known as the Orange Book.
- But by the early 1990s, it was clear that TCSEC was not viable for the new world of client/server computing.
- Its main problem was that it was not accommodating to new computing paradigms.
- And with that, TCSEC as it was known is dead.
- The very last C2 and B1 Orange Book evaluations performed by the NSA under the Orange Book itself were completed and publicly announced at the NISSC conference in October 2000.
- The C2 and B1 classes have been converted to protection profiles under the Common Criteria, however, and C2 and B1 evaluations are still being performed by commercial laboratories under the Common Criteria.
- According to the TPEP website, NSA is still willing to perform Orange Book evaluations at B2 and above, but most vendors prefer to evaluate against newer standards cast as Common Criteria protection profiles.
- Another subtle point is that the Orange Book and the Common Criteria are not exactly the same types of documents.
- Whereas the Orange Book is a set of requirements that reflect the practice and policies of a specific community, the Common Criteria are policy-independent and can be used by many organizations to articulate their security requirements.

- In Europe, the Information Technology Security Evaluation Criteria (ITSEC), already in development in the early 1990s, were published in 1991 by the European Commission.
- This was a joint effort with representatives from France, Germany, the Netherlands, and the United Kingdom contributing. Simultaneously, the Canadian government created the Canadian Trusted Computer Product Evaluation Criteria as an amalgamation of the ITSEC and TCSEC approaches.
- In the United States, the draft of the Federal Criteria for Information Technology Security was published in 1993, in an attempt to combine the various methods for evaluation criteria.
- With so many different approaches going on at once, there was a consensus to create a common approach.
- At that point, the International Organization for Standardization (ISO) began to develop a new set of standard evaluation criteria for general use that could be used internationally.
- The goal was to unite the various international and diverse standards into new criteria for the evaluation of information technology products.
- This effort ultimately led to the development of the Common Criteria, now an international standard in ISO 15408:1999.
- The official name of the standard is the International Common Criteria for Information Technology Security.

Common Criteria Sections:

Common Criteria is a set of three distinct but related parts.

These are the three parts of the Common Criteria:

Part 1:

- It is the introduction to the Common Criteria.
- It defines the general concepts and principles of information technology security evaluation and presents a general model of evaluation.
- It also presents the constructs for expressing information technology security objectives, selecting and defining information technology security requirements, and writing high-level specifications for products and systems.
- In addition, the usefulness of each part of the Common Criteria is described in terms of each of the target audiences.

Part 2:

- It details the specific security functional requirements and details a criterion for expressing the security functional requirements for Targets of Evaluation (TOE).

Part 3:

- It details the security assurance requirements and defines a set of assurance components as a standard way of expressing the assurance requirements for TOE.
- It lists the set of assurance components, families, and classes and defines evaluation criteria for protection profiles (PPs).
- A protection profile is a set of security requirements for a category of TOE and security targets (STs).
- Security targets are the set of security requirements and specifications to be used as the basis for evaluating an identified TOE.
- Part 3 also presents evaluation assurance levels that define the predefined Common Criteria scale for rating assurance for a TOE, namely the evaluation assurance levels (EALs)

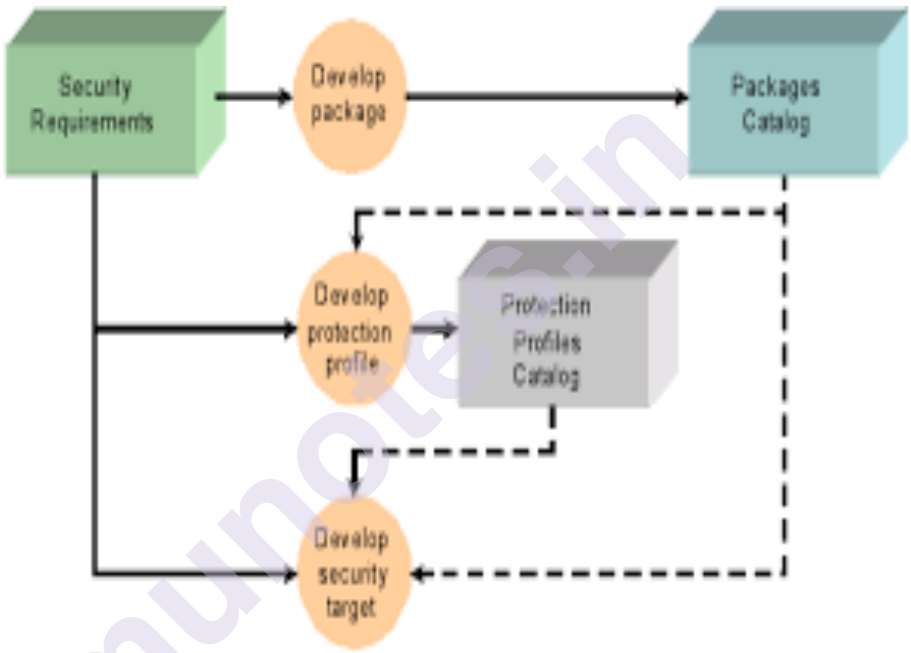
Protection Profiles and Security Targets:

- Protection profiles (PPs) and security targets (STs) are two building blocks of the Common Criteria.
- A protection profile defines a standard set of security requirements for a specific type of product (for example, operating systems, databases, or firewalls).
- These profiles form the basis for the Common Criteria evaluation.
- By listing required security features for product families, the Common Criteria allow products to state conformity to a relevant protection profile.
- During Common Criteria evaluation, the product is tested against a specific PP, providing reliable verification of the product's security capabilities.
- The overall purpose of Common Criteria product certification is to provide end users with a significant level of trust.
- Before a product can be submitted for certification, the vendor must first specify a security target.
- The security target description includes an overview of the product, potential security threats, detailed information on the implementation

of all security features included in the product, and any claims of conformity against a PP at a specified EAL.

- The vendor must submit the ST to an accredited testing laboratory for evaluation.
- The laboratory then tests the product to verify the described security features and evaluate the product against the claimed PP.
- The end result of a successful evaluation includes official certification of the product against a specific protection profile at a specified evaluation assurance level.

Figure: Common criteria modular component hierarchy



Problems with the Common Criteria:

Although there are benefits to the Common Criteria, there are also problems with this approach. The point of this section is not to detail those problems, but in a nutshell, to give you a brief summary of the issues:

Administrative overhead:

The overhead involved with gaining certification takes a huge amount of time and resources.

Expense:

Gaining certification is extremely expensive.

Labor-intensive certification:

The certification process takes months.

Need for skilled and experienced analysts:

Availability of information security professionals with the required experience is still lacking.

Room for various interpretations:

The Common Criteria leave room for various interpretations of what the standard is attempting to achieve.

A paucity of Common Criteria testing laboratories:

There are only seven laboratories in the United States.

Length of time to become a Common Criteria testing laboratory:

Even for those organizations that are interested in becoming certified, the process in and of itself takes quite a while.

10.2 LET US SUM UP

We explored different security models for operating systems, including the classics:

- Bell-LaPadula
- Biba
- Clark-Wilson
- TCSEC.
- These classic models ultimately led to today's operating system security standards.
- We also saw how the security reference monitor is a critical aspect of the underlying operating system's security functionality.
- Because all security functionality is architected, specified, and detailed in the operating system, it is the foundation of all security above it.
- Understanding how this functionality works, and how it is tied specifically to the operating system used within your organization, is crucial to ensuring that information security is maximized.
- Finally, we discussed the Trustworthy Computing initiative, international standards for operating system security, and the Common Criteria—its origins, sections, protection profiles, security targets, and shortcomings.

10.3 QUESTIONS

1. Explain the vulnerabilities of TCP/IP protocol.
2. What is the main functionality of the Access Control List?

3. Explain the role of MAC and DAC Lists.
4. Explain the different security models.
5. Explain the functionalities of Trusted Computer System Evaluation Criteria.
6. Explain the significance of the Reference Monitor.
7. Explain the significance of Bill gates' initiative TWC.
8. Explain the common criteria for Information Technology Security Evaluation of International Standards for Operating System Security.

10.4 BIBLIOGRAPHY

- Bach, Maurice. The Design of the UNIX Operating System. Prentice Hall, 1986.
- Tanenbaum, Andrew, and Albert Woodhull. Operating Systems Design and Implementation. Prentice Hall, 2006
- Principles of Computer Security: CompTIA Security+ and Beyond, Principles of Computer Security: CompTIA Security+ and Beyond, McGraw Hill, Second Edition, 2010

10.5 REFERENCES

- https://www.researchgate.net/publication/228409741_Issues_of_Operating_Systems_Security
- https://ieeexplore.ieee.org/document/5072066
- https://ieeexplore.ieee.org/document/1342833/similar#similar

*****

UNIT - V

11

VIRTUAL MACHINES AND CLOUD COMPUTING

Unit Structure

11.0 OBJECTIVES

Traditionally, applications have run directly on an operating system (OS) on a personal computer (PC) or on a server. Each PC or server would run only one OS at a time. A single application would be written many times so as to run on different OS/platforms, this created a big overhead on the part of the application designer for adding new features, testing, and marketing and also consumed much time and money.

One effective strategy for dealing with this problem is known as shareware virtualization. Virtualization technology enables a single PC or server to simultaneously run multiple operating systems or multiple sessions of a single OS. A machine running virtualization software can host numerous applications, including those that run on different operating systems, on a single hardware platform. In essence, the host operating system can support a number of virtual machines (VMs), each of which has the characteristics of a particular OS and, in some versions of virtualization, the characteristics of a particular hardware platform.

11.1 VIRTUAL MACHINES

The solution that enables virtualization is a virtual machine monitor (VMM), or commonly known today as a hypervisor. This software sits between the hardware and the VMs acting as a resource broker. The hypervisor allows multiple VMs to safely coexist on a single physical server host and share that host's resources.

VMs also require the same level of security settings as that is applicable to Windows and Unix-based systems, apart from it that the data storage security must also be applied if VMs are utilizing it.

The virtual environment apart from the VMs must also be protected, VMs are vulnerable to all the security attacks which are faced by physical servers hence they require additional protection.

11.1.1 Protecting the Hypervisor:

The hypervisor is responsible for managing all guest OS installations on a VM server, therefore any compromise of the hypervisor can cause significant damage. It would effectively allow all security controls on the virtual servers to be bypassed.

With respect to protecting the Hypervisor following must be considered:

1) Hypervisor and service console servers need to be properly patched and secured.

2) Hypervisor and service console servers must also be logically separated through the use of isolated networks with strict access controls.

3) The administration interfaces should reside on a network separate from the virtual machines.

4) Firewalls should be used to block access attempts from the virtual machines to the management consoles.

5) Administrative access to the hypervisor should be strictly controlled else an attacker will gain too much control over all the VMs.

6) Supervisory accounts for the hypervisor must get the same level of protection as privileged accounts for server and network administrator use.

7) The administrative account must not only be password protected but must have an additional way of authentication.

8) Physical access to the machine hardware must be restricted.

9) The number of administrators and their privileges must be limited.

10) Hypervisor administrators should not use the same privileged accounts they also use to manage VMs and other systems.

11) A trusted third party must perform a periodic review of administrator activities.

11.1.2 Protecting the Guest OS:

The two most commonly implemented techniques for protecting the guest OSs are:

1) Partitioning

2) Introspection

1) Partitioning:

Partitioning is considered an important security measure. The hypervisor manages access to hardware resources and each OS gets a share of its resource such as CPU, memory and storage, no OS can get access to the resources allocated to other guest OSs. This characteristic is known as partitioning, it is designed to protect each guest OS from other guest OS instances, so attacks and malware are unable to cross over.

Partitioning also reduces the threat of side-channel attacks that take advantage of hardware usage characteristics to crack encryption algorithms.

2) Introspection:

An attack referred to as escape occurs if an attacker attempts to break out of a guest OS to access the hypervisor or neighboring guest OS.

If the attacker is successful to escape and access the hypervisor then the attacker would take control over all the hypervisor's guest OS.

The hypervisor monitors and tracks the state of its guest OS, which is a function commonly, referred to as introspection.

Introspection can be integrated with intrusion detection systems (IDS) or intrusion prevention systems (IPS) and security information and event management (SIEM), to identify and alert when escape attempts occur.

11.1.3 Protecting Virtual Storage:

Guest OS systems can utilize virtual or physical network-attached storage (NAS) and storage area networks (SAN) allocated by the hypervisor to meet data storage requirements as if these storage devices were directly attached to the system. Protection in this area is focused on providing secure and controlled access to files on the virtual hard drive.

11.1.4 Protecting Virtual Networks:

The hypervisor can present the guest OS with either physical or virtual network interfaces.

Hypervisors provide three choices for network configurations:

1) **Network bridging:** The guest OS has direct access to the actual physical network interface cards (NIC) of the real server hardware.

2) **Network Address Translation (NAT):** The guest OS has virtual access to a simulated physical NIC that is connected to a NAT emulator by the hypervisor.

3) **Host-only networking:** A guest OS has virtual access to a virtual NIC that does not actually route to any physical NIC.

With respect to security in such situations, security devices, such as IDS or IPS, can monitor and control network traffic using network bridging and NAT and, to a lesser extent, host-only networking.

In the case of host-only networking, introspection can be used to compensate for this lack of visibility.

Network segmentation is one of the best security practices irrespective of environment.

11.2 CLOUD COMPUTING

There is an increasingly prominent trend in many organizations to move a substantial portion or even all information technology (IT) operations to an Internet-connected infrastructure known as enterprise cloud computing.

Also at the same time, individual users of PCs and mobile devices are relying more and more on cloud computing services to backup data, synch devices, and share.

Cloud Computing can be defined as "A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

11.2.1 Types of Cloud Services:

The following are the most common types of services with which we find the term cloud associated

1) Software as a Service (SaaS):

SaaS cloud provides service to customers in the form of software, specifically application software running on and accessible in the cloud.

2) Platform as a Service (PaaS):

A PaaS cloud provides service to customers in the form of a platform on which the customer's applications can run.

3) Infrastructure as a Service (IaaS):

IaaS offers the customer processing, storage, networks, and other fundamental computing resources so that the customer can deploy and run arbitrary software, which can include operating systems and applications

4) Communications as a Service (CaaS):

The integration of real-time interaction and collaboration services to optimize business processes. This service provides a unified interface and consistent user experience across multiple devices.

5) Data Storage as a Service (DSaaS):

The provision and use of data storage and related capabilities. DSaaS describes a storage model where the client leases storage space from a thirdparty provider.

6) Network as a Service (NaaS):

NaaS involves the optimization of resource allocations by considering network and computing resources as a unified whole. NaaS can include flexible and extended virtual private network (VPN), bandwidth on demand, custom routing, multicast protocols, security firewall, intrusion detection and prevention, wide-area network (WAN), content monitoring and filtering, and antivirus.

7) Database as a Service:

Database functionalities are on demand where the installation and maintenance of the databases are performed by the cloud service provider.

8) Desktop as a Service:

The ability to build, configure, manage, store, execute, and deliver user desktop functions remotely.

9) E-mail as a Service:

A complete e-mail service, including related support services such as storage, receipt, transmission, backup, and recovery of e-mail.

11.2.2 Cloud Computing Security Benefits:

Cloud computing can provide a higher level of security as compared to traditional computing environments. A properly designed cloud computing infrastructure offers better physical and operational security controls at lower costs.

Some of the services are:

1) Centralized data:

Data leakage through laptop data loss and backup tape loss could be reduced by cloud computing.

2) Monitoring:

Centralized storage is easier to control and monitor.

3) Forensics and incident response:

A dedicated forensic server can be built in the same cloud as the corporate servers but placed offline, ready to be used, and brought online as required.

4) Password assurance testing:

For organizations that routinely crack passwords to check for weaknesses, password cracking times can be significantly decreased.

5) Logging:

Effectively unlimited storage for logging, with reduced concerns about insufficient disk space being allocated for system logging.

6) Testing security changes:

Vendor updates and patches, as well as configuration changes for security purposes, can be applied using a cloned copy of the production server, with low-cost impact testing and reduced start-up time.

7) Security infrastructure:

SaaS providers that offer security technologies to customers share the costs of those technologies among their customers who use them.

11.2.3 Security Considerations:

1) While considering the security aspects, private data and public data must be separated.

2) Private data requires strict security controls as compared to public data.

3) Organizations must make a slow transition to the cloud rather than do it all at a time.

4) Organizations that are currently leveraging cloud computing to streamline their business processes and systems so they can minimize the amount of integration needed to use cloud platforms are realizing the greatest benefits today.

5) Public clouds are accessed over the Internet and face bandwidth limitations hence scaling to larger Internet bandwidths can significantly increase the overall ownership cost of cloud solutions.

6) Review the potential cost savings of cloud environments.

7) Knowing and controlling the location of data is important for many reasons.

8) If the data servers are in hostile nations, then it can cause security concerns.

9) For sensitive and private data, colocation is also a concern. The sensitive data must be logically separated.

10) Any sensitive or confidential information placed into a cloud environment should be protected beyond the security features of the cloud service itself.

11.2.4 Cloud Computing Risks and Remediation:

Cloud together with data centers raises important issues and concerns, which must be addressed.

1) Availability:

The availability issue in Cloud services can be managed by using redundant service providers so a failure at one provider will not result in a loss of service.

2) Data persistence:

This is concerned with what happens to the data when it is deleted from the cloud.

3) Patriot Act ramifications:

Some countries like the US impose their right to monitor and capture all traffic from a service provider on demand.

4) Compliance ramifications:

Some government regulations do not allow cloud computing.

5) PCI compliance:

Requires that you know and can demonstrate exactly where and on what physical server your data resides.

6) Migration:

Physical-to-cloud and cloud-to-physical capability may be required to move data into the cloud from the local computing environment, or vice versa.

7) Confidentiality:

The responsibility for controlling data in a cloud environment is shared between the cloud provider and the customer. Any data that an organization feels is confidential must be housed in a private network or private cloud, not in a public cloud.

11.2.5 Cloud Computing Security Incidents:

The following table shows some of the security incidents recorded by websites run by a security professional.

Provider	Incident Type	Incident Subtype	Affected	Notes
Apple	Outage	Disaster recovery	All	Full extended outage
Google	Outage	Change management	Many	Users unable to use webmail due to issues with loading contacts between

				14:00 and 16:00 PT
Nirvanix MediaMax	Data loss	Closure	20,000	Data claimed to be safe but inaccessible.
FlexiScale AWS	Outage	Design fault	All	Full outage for eight (weekend) hours.
Apple	Outage	Migration	All	The Scheduled outage window exceeded during the upgrade to MobileMe

11.2.6 Cloud Security Technologies:

Cloud computing providers offer several security services to remediate some of the risks inherent to the cloud environment.

Some of them include the following:

1) Communication encryption
2) File-system encryption
3) Auditing
4) Traditional network firewalls
5) Application firewalls
6) Content filtering
7) Intrusion detection
8) Geographic diversity

11.2.7 Vendor Security Review:

A third-party security review must be performed to validate the security practices of cloud providers.

The following attributes must be reviewed:

1) Physical security
2) Backups and/or data protection
3) Administrator access
4) Firewalls
5) Hypervisor security
6) Customer and instance isolation

7) Intrusion detection and anomaly monitoring

8) Data transmission security

9) Data storage security

11.3 RISK AND REMEDIATION ANALYSIS

The risks associated with cloud computing include the set of risks associated with traditional data centers combined with those of Internet-based services, added to a new set of risks that arise from the convergence of private and public environments.

The following categories of risks are divided according to the classic "CIA" triad of Confidentiality, Integrity, and Availability.

Within each identified risk, the three Ds of security (Defence, Detection, and Deterrence) are applied accordingly.

11.3.1 Confidentiality Risks:

These risks are associated with vulnerabilities and threats concerned with privacy and control of information. It is required to make the information available in a controlled fashion to only those parties that need it, without exposing it to unauthorized parties.

Data leakage, theft, exposure, and forwarding:

The loss of information such as customer data and other types of intellectual property through intentional or unintentional means.

There are four major threat vectors for data leakage:

1. Theft by outsiders
2. Malicious sabotage by insiders
3. Inadvertent misuse by authorized users and
4. Mistakes created by unclear policies.

i) Defence:

Employ software controls to block inappropriate data access through a data loss prevention (DLP) solution. Avoid placing sensitive, confidential, or personally identifiable (PII) information in the cloud.

ii) Detection:

Use water-marking and data classification labeling along with monitoring software to track data flows.

iii) Deterrence:

Establish clear and strong language in contractual agreements with service providers that specify how data privacy will be enforced and maintained.

iv) Residual risk:

Data persistence within the cloud vendor environment in relation to multiple untraceable logical disk storage locations and vendor administrative access that exposes private data to administrators.

Espionage, packet sniffing, packet replay:

The unauthorized interception of network traffic for the purpose of gaining information intentionally, using tools to capture network packets or tools to reproduce traffic and data that was previously sent on a network.

i) Defence:

Encrypt data at rest as well as data in transit through the use of strong encryption technologies for file encryption (e.g., PGP), as well as network encryption between servers and over the Internet (e.g., TLS, SSL, SFTP). Preference should be given to cloud providers that offer link-layer data encryption.

ii) Detection:

Not much can be done today to find out when somebody has intercepted your data; however, an IDS capability can help to identify anomalous behavior on the network that may indicate unauthorized access attempts.

iii) Deterrence:

Transfer the risk of unauthorized access to the service provider using specific contract language.

iv) Residual risk:

Data can be stolen from the network through tools that take advantage of network topologies, network weaknesses, compromised servers and network equipment, and direct access to network devices.

Inappropriate administrator access:

Using privilege for access privileges levels generally reserved for system administrators that provide full access to a system and all data that system has access to, in order to view data or make changes without going through the system's authorization processes. Administrators have the capability of bypassing all security controls, and this can be used to intentionally or mistakenly compromise private data.

i) Defense:

Minimize the number of service provider administrators for each cloud service function. Perform a vendor security review to validate these practices before engaging or signing with a cloud vendor.

ii) Detection:

Review the cloud provider's administrative access logs for their internal infrastructure on a monthly or quarterly basis. Review the provider's list of administrators on a biannual basis.

iii) Deterrence:

Select only those cloud providers that can demonstrate robust system and network administration practices that are also willing to agree with customer conditions.

iv) Residual risk:

Because administrators have full control, there is a possibility that they will intentionally or accidentally abuse their access privileges, resulting in the compromise of personal information or service availability.

Storage persistence:

Data may remain on a hard drive long after it is no longer required and also potentially after it has been deleted. As this data may be deleted but not strongly overwritten, it is at an increased risk of future data recovery by unauthorized individuals.

i) Defence:

Insist that vendors maintain a program that includes Department of Defence (DoD) disk wiping when disks are replaced or reallocated. Dead disks should be degaussed or destroyed to prevent data disclosure.

ii) Detection:

Not much can be done to find out when data persists on a disk that has been taken offline.

iii) Deterrence:

Establish disk wiping practices before selecting a vendor and ensure that contract language clearly establishes these requirements.

iv) Residual risk:

Data can remain on physical media long after it has been deleted.

Storage platform attacks:

Direct attacks against a SAN or storage infrastructure, including the use of a storage system's management control, can provide access to private data, bypassing the controls built into an operating system because the operating system is out of the loop.

i) Defence:

Ensure that vendors have implemented strong compartmentalization and role-based access control on their storage systems that access to the management interface of vendor storage systems is not accessible through the customer network.

ii) Detection:

Implement IDS for the storage network and review storage system access control logs on a quarterly basis.

iii) Deterrence:

Ensure that the cloud service provider has strong legal representation and a commitment to identifying and prosecuting attackers.

iv) Residual risk:

Data can be stolen directly from the SAN and you may find out about it after the fact or not at all.

Misuse of data:

People who are authorized to access data also have the opportunity to do anything with that data, including actions that they are not permitted to perform. Examples include employees who leak information to competitors, developers who perform testing with production data, and people who take data out of the controlled environment of the organization's private network into their unprotected home environment.

i) Defence:

For employees, use security controls similar to those in private data networks, such as DLP, role-based access controls, and scrambling of test and development data. Block the ability to send e-mail attachments to external e-mail addresses.

ii) Detection:

Use water-marking and data classification labelling along with monitoring software to track data flows.

iii) Deterrence:

Use a security awareness program along with penalties and sanctions to deter people from transferring data from a controlled environment to an uncontrolled environment.

iv) Residual risk:

People can find ways around controls to put data into uncontrolled environments where it can be stolen or misused.

Fraud:

Illegally (or deceptively) gaining access to information that a person is not authorized to access. Fraud can be perpetrated by outsiders but is usually performed by trusted employees.

i) Defence:

Use checks and balances along with sufficient separation of duties to reduce the dependence on single individuals. Ensure that business processes include management reviews and approvals.

ii) Detection:

Perform regular audits on computing system access and data usage with special attention to unauthorized access.

iii) Deterrence:

For employees, ensure that there is a suitable penalty process. For service providers, transfer risks through the use of contractual language.

iv) Residual risk:

Fraudulent practices can result in significant reputation and financial damages.

Hijacking:

The exploitation of a valid session to gain unauthorized access to information or services in a computer system, in particular, the theft of a magic cookie used to authenticate a user to a remote server.

Any protocol in which a state is maintained using a key passed between two parties is vulnerable, especially if it's not encrypted. This also applies to the cloud environment's management credentials; if the entire cloud service is managed using session keys, the entire environment can be taken over through the effective use of a session hijacking attack.

i) Defence:

Look for solid identity management implementations from service providers that specifically address this risk using strong, non-guessable session keys with encryption. Use good key management processes and

practices and key escrow and key recovery practices as a customer so employee departures do not result in the inability to manage your service.

ii) Detection:

Routinely monitor logs for access to cloud resources and their management interface to identify unexpected behavior.

iii) Deterrence:

Not much can be done to deter attackers from hijacking sessions outside of aggressive legal response.

iv) Residual risk:

Attackers can impersonate valid users of cloud services or even use administrative credentials to lock you out or damage your entire infrastructure.

11.4 INTEGRITY RISKS

Any change in the information intentionally or unintentionally can cause integrity risks. These risks affect the validity of information and the assurance that the information is correct. If information can be changed without warning, authorization, or an audit trail, its integrity cannot be guaranteed.

Malfunctions:

Computer and storage failures can cause data corruption.

i) Defence:

Make sure the service provider you select has appropriate RAID redundancy built into its storage network and that creating archives of important data is part of the service.

ii) Detection:

Employ integrity verification software that uses checksums or other means of data verification.

iii) Deterrence:

Owing to the nature of the data and the fact that there is no human interaction, little can be accomplished.

iv) Residual risk:

Technology failures that damage data may result in operational or compliance risks.

Data deletion and data loss:

Accidental or intentional destruction of any data, including financial, company, personal, and audit trail information. Destruction of data due to computer system failures or mishandling.

i) Defense:

In the cloud environment, ensure that your critical data is redundantly stored and housed with more than one cloud service provider.

ii) Detection:

Maintain and review audit logs that relate to data deletion.

iii) Deterrence:

Maintain education and awareness programs for individuals who access and manage data. Ensure that appropriate data owners are assigned who have full authority and control over data.

iv) Residual risk:

Once critical data is gone, if it can't be restored it is gone forever.

Data corruption and data tampering:

Changes to data are caused by a malfunction in computer or storage systems, or by malicious individuals or malware. Modification of data with intent to defraud.

i) Defense:

Cloud services offer virtually unlimited data storage, hence we can keep virtually unlimited copies of prior versions. All virtual servers must be protected by antivirus (AV) software.

ii) Detection:

Use integrity-checking software to monitor and report on any alteration of key data.

iii) Deterrence:

Maintain education and awareness programs for individuals who access and manage data. Ensure that suitable data owners are assigned who have authority and control over data.

iv) Residual risk:

Corrupted or damaged data can cause significant issues because valid, reliable data is the cornerstone of any computing system.

Accidental modification:

This is the most common cause of data integrity loss, changes made to data either because the individual thought he or she was modifying something else or because of incorrect input.

i) Defense:

Since cloud services offer virtually unlimited data storage hence we can store and maintain virtually unlimited copies of prior versions. Ensure that all virtual servers are protected by AV software. Maintain role-based access control to all data based on the least privilege principle.

ii) Detection:

Use integrity-checking software to monitor and report on alterations to key data.

iii) Deterrence:

Maintain education and awareness programs for individuals who access and manage data. Ensure that appropriate data owners are assigned who have full authority and control over data.

iv) Residual risk:

Corrupted or damaged data can cause significant issues because valid, reliable data is the cornerstone of any computing system.

Phishing:

Often come through e-mail, the act of tricking a victim into giving out personal information is a common tactic of social engineering.

i) Defense:

Employ anti-phishing technologies to block rogue websites and detect false URLs. Use multifactor authentication for customer-facing systems to ensure that users are aware when they are redirected to fake copies of your websites. Send periodic informational updates and educational materials to customers explaining how the system works and how to avoid phishing. Never send e-mails to customers that include or request personal details, including customer IDs or passwords.

ii) Detection:

Use an application firewall to detect when remote sites are trying to copy or emulate your website.

iii) Deterrence:

Maintain education and awareness programs for individuals who use and store personal information about employees or customers.

iv) Residual risk:

Significant reputation risk owing to exposure in the public media or allegations of personal data loss commensurate with the business risks of losing backup tapes or a compromise of a database containing customer information. Bad publicity can lead to both long and short-term loss of corporate reputation.

11.5 AVAILABILITY RISKS

These risks are associated with vulnerabilities and threats pertaining to the reliability of services, given the need to use services reliably with low risk and incidence of an outage.

Denial of service:

A denial of service (DoS) attack or distributed denial of service (DDoS) attack is an attempt to make a computer resource unavailable to its intended users. Cloud services can be especially vulnerable to volumetric DDoS attacks, in which large numbers of computers flood the cloud networks and servers with more data than they can handle, causing them to grind to a halt.

i) Defense:

Select a service provider that has solid protection against network-based attacks. Implement firewalls and network filtering at the network perimeter of the cloud infrastructure (primarily the Internet access point) to block attacks and hostile networks using a network blacklist. In addition, use redundant providers because an attack against one provider's environment may not affect another.

ii) Detection:

Select a service provider that performs and monitors intrusion detection on a 24×7 basis and sign up for any appropriate additional services relating to this capability.

iii) Deterrence:

Work with the service provider's legal department to ensure that attackers are found and prosecuted.

iv) Residual risk:

As most DoS attacks originate from other countries and can be hard to detect and track, there is little that you can do about the ones that get through an environment's defenses.

Outage:

Any unexpected downtime or unreachability of a computer system or network.

i) Defense:

The primary defense against any service outage is redundancy. Ensure that environments can be automatically switched to a different provider during an outage.

A solid disaster recovery plan must be ready for extended outages.

ii) Detection:

Employ monitoring tools to monitor the availability and response time of the cloud environment continuously.

iii) Deterrence:

Outages are expensive. Calculate the cost of downtime and make sure the contract with the service provider allows compensation for real costs incurred, not just remuneration for the cost of the service itself.

iv) Residual risk:

Because outages generally occur because of software problems, little can be done to stop them from happening.

Instability and application failure:

Loss of functionality or failure of a computer or network owing to problems (bugs) in the software or firmware. Freezing, locking, or crashing of a program causing unresponsiveness.

i) Defence:

Ensure that the vendor applies all software updates for its infrastructure on a frequent basis. Do the same for all customer-owned virtual systems.

ii) Detection:

Implement service monitoring to detect and alert when an application does not respond correctly.

iii) Deterrence:

Use legal language to clearly set the expectation that the service provider will maintain a stable environment.

iv) Residual risk:

As the instability of applications and infrastructure generally occurs as a result of a software problem, little can be done to stop them from occurring.

Slowness:

Unacceptable response time of a computer or network.

i) Defence:

Using redundant providers and Internet connections set up the architecture so application access will automatically switch to the fastest environment. Also, ensure service providers have implemented high-capacity services with automatic expansion of resources.

ii) Detection:

Monitor the response time of applications on a continuous basis and ensure that alerts have an out-of-band path to support staff so response problems don't stop alerts from being delivered.

iii) Deterrence:

Establish contract language with service providers that provide penalties in the form of compensation to you for unacceptable response times.

HA failure:

The discovery is a device that was supposed to fail over doesn't actually take over when it should.

i) Defence:

Monitor the health of secondary systems or all systems in an HA cluster.

ii) Detection:

Perform periodic failover testing.

iii) Deterrence:

Not much can be done from a service provider perspective to guarantee that customer systems will switch over when they are supposed to.

iv) Residual Risk:

Sometimes a primary device slows down to the point that it becomes unresponsive for all practical purposes, but because it's not officially "down" according to the software, the backup system doesn't take over.

Backup failure:

The discovery that those data backups you were relying on aren't actually any good.

i) Defence:

Leverage provider elasticity to avoid the use of traditional offline (tape or optical) backups.

ii) Detection:

Frequently perform recovery testing to validate the resilience of data.

iii) Deterrence:

Establish a data-loss clause in the contract with the service provider so they are obligated to assist with unforeseen data loss.

iv) Residual risk:

Backups fail, but multiple recovery paths can eliminate most of the risk.

Secure Application Design:

Applications such as web applications, client applications, and remote administration are designed for some purpose and to run in an environment.

These applications after being deployed in their original form could defend themselves from threats, mistakes, or misuse. A malicious user can exploit the vulnerability of the applications and launch an attack.

Such attacks would eventually lead to customers who are unhappy with their software vendors, regardless of whether or not the customers were willing to pay for a security before the incident occurred.

Therefore, security is becoming more important to organizations that produce software, and building security into the software upfront is easier (and cheaper) than waiting until the software is already out in the field and then providing security updates.

While the deployment environment can help protect the application to some extent, every application must be secure enough to protect itself from whatever meaningful attacks the deployment environment cannot prevent, for long enough for the operator to notice and respond to attacks in progress.

11.6 SECURE DEVELOPMENT LIFECYCLE

A secure development lifecycle (SDL) is essentially a development process that includes security practices and decision-making inputs.

A typical SDL actually affects two to three lifecycles, the specifics of which vary by organization:

1. **The application lifecycle:** In which an application begins as an idea and then is planned, designed, developed, tested, documented, released, sometimes deployed and operated, maintained, and eventually ended.

2. **The employee lifecycle:** In which an employee is selected, hired, brought on board, changes job responsibilities, and eventually leaves the organization.

3. **The project or contract lifecycle:** If any development is outsourced, in which a contract is negotiated, results are accepted, and vendors are paid.

The SDL itself is created, operated, measured, and changed over time following a business process lifecycle.

Typically, an SDL contains three primary elements:

1. Security activities that don't exist at all in the original lifecycle; for instance, threat modeling.
2. Security modifications to existing activities; for instance, adding security checks to existing peer reviews of code.
3. Security criteria that should affect existing decisions; for instance, the number of open high-severity security issues when a decision to ship is made.

Adding security is cheapest if it is included from the beginning of the lifecycle.

11.7 APPLICATION SECURITY PRACTICES

Most secure development lifecycles contain the following practices and decisions in various forms.

11.7.1 Security Training:

A security training program for development teams includes technical security awareness training for everyone and role-specific training for most individuals. Role-specific training goes into more detail about the security activities a particular individual participates in, and the technologies in use (for developers).

11.7.2 Secure Development Infrastructure:

At the beginning of a new project, source code repositories, file shares, and build servers must be configured for team members' exclusive access, bug tracking software must be configured to disclose security bugs only according to organization policies, project contacts must be registered in case any application security issues occur, and licenses for secure development tools must be acquired.

11.7.3 Security Requirements:

Security requirements may include access control matrices, security objectives, abuse cases, references to policies and standards, logging requirements, security bug bars, assignment of a security risk or impact level, and low-level security requirements such as key sizes or how specific error conditions should be handled.

11.7.4 Secure Design

Secure design activities usually revolve around secure design principles and patterns. They also frequently include adding information about security properties and responsibilities to design documents.

11.7.5 Threat Modeling:

Threat modeling is a technique for reviewing the security properties of a design and identifying potential issues and fixes. Architects can perform it as a secure design activity, or independent design reviewers can perform it to verify architects' work.

11.7.6 Secure Coding:

Secure coding includes using safe or approved versions of functions and libraries, eliminating unused code, following policies, handling data safely, managing resources correctly, handling events safely, and using security technology correctly.

11.7.7 Security Code Review:

To find security issues by inspecting application code, development teams may use static analysis tools, manual code review, or a combination. Static analysis tools are very effective at finding some kinds of mechanical security issues but are usually ineffective at finding algorithmic issues like incorrect enforcement of business logic. Manual code review by someone other than the code author is more effective at finding issues that involve code semantics but requires training and experience. Manual code review is also time-consuming and may miss mechanical issues that require tracing large numbers of lines of code or remembering many details.

11.7.8 Security Testing:

To find security issues by running application code, developers and independent testers perform repeatable security testing, such as fuzzing and regression tests for past security issues, and exploratory security testing, such as penetration testing.

11.7.9 Security Documentation:

When an application will be operated by someone other than the development team, the operator needs to understand what security the application needs the deployment environment to provide, what settings can affect security, and how to handle any error messages that have a security impact. The operator also needs to know if a release fixes any vulnerabilities in previous releases.

11.7.10 Secure Release Management:

When an application will be shipped, it should be built on a limited-access build server and packaged and distributed in such a way that the recipients can verify it is unchanged. Depending on the target platform, this may mean code signing or distributing signed checksums with the binaries.

11.7.11 Dependency Patch Monitoring:

Any application that includes third-party code should monitor that external dependency for known security issues and updates, and issue a patch to update the application when any are discovered.

11.7.12 Product Security Incident Response:

Product security incident response includes contacting people who should help respond, verifying and diagnosing the issue, figuring out and implementing a fix, and possibly managing public relations. It does not usually include forensics.

11.7.13 Decisions to Proceed:

Any decision to ship an application or continue its development should take security into account. At ship time, the relevant question is whether the application can be reasonably expected to meet its security objectives. It means that security validation activities have occurred and no critical or high-severity security issues remain open.

11.8 WEB APPLICATION SECURITY

Web application security includes understanding the vulnerabilities attackers can exploit in insecure web applications and compromise a web server or deface a website, and how developers can avoid introducing these vulnerabilities.

The several web application security concerns to be considered are as follows:

1. SQL injection
2. Forms and scripts
3. Cookies and session management
4. General attacks

11.8.1 SQL injection:

SQL injection is a technique to inject crafted SQL into user input fields that are part of web forms—it is mostly used to bypass custom logins to websites. However, SQL injection can also be used to log in to or even to take over a website, so it is important to secure against such attacks.

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

Example 1:

In some situations, an attacker can escalate an SQL injection attack to compromise the underlying server or other back-end infrastructure or perform a denial-of-service attack.

Consider a shopping application that displays products in different categories. When the user clicks on the Gifts category, their browser requests the URL:

https://insecure-website.com/products?category=Gifts

This causes the application to make an SQL query to retrieve details of the relevant products from the database:

SELECT * FROM products WHERE category = 'Gifts' AND released = 1

This SQL query asks the database to return:

all details (*)

from the products table

where the category is Gifts

and released is 1.

The restriction released = 1 is being used to hide products that are not released. For unreleased products, presumably released = 0.

The application doesn't implement any defenses against SQL injection attacks, so an attacker can construct an attack like:

https://insecure-website.com/products?category=Gifts'--

This results in the SQL query:

SELECT * FROM products WHERE category = 'Gifts'--' AND released = 1

The key thing here is that the double-dash sequence -- is a comment indicator in SQL, and means that the rest of the query is interpreted as a comment. This effectively removes the remainder of the query, so it no longer includes AND released = 1. This means that all products are displayed, including unreleased products.

Going further, an attacker can cause the application to display all the products in any category, including categories that they don't know about:

https://insecure-website.com/products?category=Gifts'+OR+1=1--

This results in the SQL query:

SELECT * FROM products WHERE category = 'Gifts' OR 1=1--' AND released = 1

The modified query will return all items where either the category is Gifts, or 1 is equal to 1. Since 1=1 is always true, the query will return all items.

Example 2:

Consider an application that lets users log in with a username and password. If a user submits the username wiener and the password

bluecheese, the application checks the credentials by performing the following SQL query:

SELECT * FROM users WHERE username = 'wiener' AND password = 'bluecheese'

If the query returns the details of a user, then the login is successful. Otherwise, it is rejected.

Here, an attacker can log in as any user without a password simply by using the SQL comment sequence -- to remove the password check from the WHERE clause of the query. For example, submitting the username administrator'-- and a blank password results in the following query:

SELECT * FROM users WHERE username = 'administrator'--' AND password = ''

This query returns the user whose username is 'administrator' and successfully logs the attacker in as that user.

Solutions for SQL Injection:

Developers and administrators can take a number of different steps in order to solve the SQL injection problem.

These are some solutions for developers:

1. Filter all input fields for apostrophes (') to prevent unauthorized logins.
2. Filter all input fields for SQL commands like insert, select, union, delete, and exec to prevent server manipulation.
3. Limit input field length (which will limit attackers' options), and validate the input length with server-side scripts.
4. Use the option to filter "escape characters" (characters that can be used to inject SQL code, such as apostrophes) if the database offers that function.
5. Place the database on a different computer than the web server. If the database is hacked, it'll be harder for the attacker to reach the web server.
6. Limit the user privileges of the server-side script.
7. Delete all unneeded extended stored procedures to limit attackers' possibilities.
8. Place the database in a separate container (behind a firewall), separated from the web container and application server.

The administrator can mitigate the risks by running some tests and making sure that the code is secure:

1. Make sure the web server returns a custom error page. This way, the server won't return the SQL error, which will make it harder for the attacker to gain data about the SQL query.
2. Deploy only web applications that separate the database from the web server.
3. Hire an outside agency to perform penetration tests on the web server and to look for SQL injection exploits.
4. Use a purpose-built automated scanning device to discover SQL injection exploits that result from programmers' mistakes.
5. Deploy security solutions that validate user input and that filter SQL injection attempts.

11.8.2 Forms and Scripts:

Forms are used to allow a user to enter input, but forms can also be used to manage sessions and to transfer crucial data within the session (such as a user or session identifier).

Attackers can exploit the data embedded inside forms and can trick the web application into either exposing information about another user or charge a lower price in e-commerce applications.

Three methods of exploiting forms are these:

1. Disabling client-side scripts
2. Passing parameters in the URLs
3. Passing parameters via hidden fields

11.8.3 Client-Side Scripts:

Some developers use client-side scripts to validate input fields in various ways:

1. Limit the size of the input fields
2. Disallow certain characters (such as apostrophes)
3. Perform other types of validation (these can be specific to each site)

By disabling client-side scripting (either JavaScript or VBScript), this validation can be easily bypassed. A developer should validate all fields on the server side. This may require additional resources on the server.

11.8.4 Passing Parameters via URLs:

The Web Parameter Tampering attack is based on the manipulation of parameters exchanged between client and server in order to modify

application data, such as user credentials and permissions, price and quantity of products, etc. Usually, this information is stored in cookies, hidden form fields, or URL Query Strings, and is used to increase application functionality and control.

This attack can be performed by a malicious user who wants to exploit the application for their own benefit or an attacker who wishes to attack a third person using a Man-in-the-middle attack. In both cases, tools like Webscarab and Paros proxy are mostly used.

The attack's success depends on integrity and logic validation mechanism errors, and its exploitation can result in other consequences including XSS, SQL Injection, file inclusion, and path disclosure attacks.

Example 1:

The parameter modification of form fields can be considered a typical example of a Web Parameter Tampering attack.

For example, consider a user who can select form field values (combo box, check box, etc.) on an application page. When these values are submitted by the user, they could be acquired and arbitrarily manipulated by an attacker.

Example 2:

When a web application uses hidden fields to store status information, a malicious user can tamper with the values stored on their browser and change the referred information. For example, an e-commerce shopping site uses hidden fields to refer to its items, as follows:

<input type="hidden" id="1008" name="cost" value="70.00">

In this example, an attacker can modify the "value" information of a specific item, thus lowering its cost.

11.8.5 Passing Data via Hidden Fields:

The post method sends the data using the POST HTTP command. Although the data does not travel in the URL, it can be exploited rather easily as well.

Consider the following form:

```
...
<form action="checkout.asp" method="post">
<input type="hidden" name="UserID" value="102" />
<p><input type="submit" name="submit" value="checkout" /></p>
</form>
...
```

This form transmits the user identifier using POST. An attacker can save the HTML, modify the UserID field, and modify the checkout.asp path (to link to the original site, like this:

<form action="http://example/checkout.asp"…),

run it (by double-clicking on the modified local version of the HTML page), and submit the modified data.

11.8.6 Solving Data-Transfer Problems:

The developer can prevent attackers from modifying data that is supposed to be hidden by managing the session information, by using GUIDs, or by encrypting the information.

1) Managing Session Information:

Most server-side scripting technologies allow the developer to store session information about the user—this is the most secure method to save session-specific information because all the data is stored locally on the web server machine.

2) Using GUIDs:

A globally unique identifier, or GUID, is a 128-bit randomly generated number that has 2128 possible values. GUIDs can be used as user identifiers by the web application programmer. With such an enormous space (2128) it would be impossible for an attacker to guess the correct GUID.

3) Encrypting Data:

The developer can pass encrypted data rather than passing the data in clear text. The data should be encrypted using a symmetric key. If an attacker tries to modify the encrypted data, the client will detect that someone has tampered with the data.

4) Cookies and Session Management:

Sessions are used to track user activities, such as a user adding items to their shopping cart, the site keeps track of the items by using the session identifier. Sessions use cookies that are stored in the user's browser.

Each time the user visits a website that sent a cookie, the browser will send the cookie back to the website.

Sessions use cookies to identify users and pair them with an active session identifier. Attackers can abuse both sessions and cookies, with various risks such as:

1) Session theft
2) Managing sessions by sending data to the user
3) Web server cookie attacks
4) Securing sessions

1) Session Theft:

Whenever a user logs into a website, the website tags the session as authenticated and allows the user to browse to secure areas for authenticated users. The website uses cookies in order to save sensitive data, but an attacker can exploit this. Server-side cookies are another alternative, suppose the website uses e-mail addresses as the identifying data. After the user has logged in, the system will send the browser a cookie containing the user's e-mail address. For every page this user will visit, the browser will transmit the cookie containing the user's e-mail address. The site checks the data in the cookie and allows the user to go where their profile permits.

An attacker could modify the data in the cookie, suppose the cookie contains smilemail@site.com. Each time we access the site we can automatically access restricted areas. If the attacker changes the e-mail address in his cookie (located on his computer) to somesmilemail@site.com, the next time the attacker accesses the site, it will think he is the user somesmilemail and allow him to access that user's data.

2) Managing Sessions without Sending Data to the User:

Some users disable cookies, which means they also don't allow session management which requires cookies. Unless the site is using the less secure get or post methods to manage sessions, the only way to keep track of users is by using their IP address as an identifier. However, this method has the following problems:

a) Some users surf through Network Address Translation (NAT), such as corporate users, and they will share one or a limited number of IP addresses.

b) Some users surf through anonymous proxies, and they will share this proxy IP address.

c) Some users use dial-up connections and share an IP address pool, which means that when a user disconnects, the next connected user will get that IP address.

3) Web Server Cookie Attacks:

a) An attacker can exhaust the resources of a web server using cookie management by opening many connections from dedicated software.

b) Each session will consume the system resources such as memory or hard drive.

c) The solution to this problem is to configure a firewall so that it does not allow more than a particular number of connections per second, which will prevent an attacker from initiating an unlimited number of connections.

4) Securing Session Tracking:

Securing the session can be done in the following ways:

a) By using a hard-to-guess identifier that is not derived from the user's data, such as an encrypted string or GUID.

b) In case of multiple users, the IP address can be tied to the identifier.

c) A short timeout can be used to delete an active session after the time limit has elapsed, to ensure that the session is closed by the server if the user does not end the session properly.

11.8.7 General Attacks:

These are those attacks that do not come under any category but pose a significant risk, some of which are as follows

1) Vulnerable scripts

2) Attempts to brute-force logins

3) Buffer overflows

1) Vulnerable Scripts:

a) Some publicly used scripts contain bugs that allow attackers to view or modify files or even take over the web server's computer.

b) The best way to find out if the web server contains such scripts is to run a vulnerability scanner.

c) If such a script is found, it should be either updated (with a non-vulnerable version) or replaced with an alternative script.

2) Brute-Forcing Logins:

An attacker can try to brute-force the login (trying all the possible ways) using a dictionary. There are a number of ways to combat brute-force attacks:

a) Limit the number of connections per second per IP address.

b) Force users to choose strong passwords that contain upper and lowercase letters and digits.

3) Buffer Overflows:

a) Buffer overflows can be used to gain control over the web server.

b) The attacker sends a large input that contains assembly code, and if the script is vulnerable, this string is executed and usually runs a Trojan that will allow the attacker to take over the computer.

11.9 CLIENT APPLICATION SECURITY

Application security is mainly controlled by the developer of the application. Writing a secure application is difficult, because every aspect of the application, like the GUI, network connectivity, OS interaction, and sensitive data management, requires extensive security knowledge in order to secure it.

The following security issues must be kept in mind while designing an application:

1) Running privileges
2) Administration
3) Application updates
4) Integration with OS security
5) Adware and spyware
6) Network access

1) Running Privileges:

An administrator must strive to run an application with the least privileges possible, it prevents the system from the following attacks

a) If the application is exploited by attackers, they will have the privileges of the application. If the privileges are low enough, the attackers won't be able to take the attack further.

b) Low privileges protect the computer from an embedded Trojan (in the application) because the Trojan will have fewer options at its disposal.

c) When an application has low privileges, the user won't be able to save data in sensitive areas or even access key network resources.

d) If an application requires administrative privileges but there is no obvious reason why it needs them, then they can be run within a sandbox.

e) Sandboxes can limit access to the registry, OS data directory, and network usage. This isolates the application from sensitive OS areas and other user-defined locations, such as those containing sensitive data.

2) Application Administration:

Most applications offer some type of interface for administration, and each administration method poses security risks that must be addressed, the interfaces used are.

a) INI/Conf file

b) GUI

c) Web-based control

a) INI/Conf Files:

This is the most basic method of administrating an application

Application can be secured in the following scenarios:

i) Local access: To secure such an application running on the local machine, the administrator needs to limit access to the configuration files by using built-in OS access management.

ii) Remote access: To secure applications in this scenario, strong authentication methods must be followed.

b) GUIs:

Most applications have a GUI for administrating them. Therefore, apart from providing security at the GUI level, the communication between the GUI and the application must also be secured. Two cases arise

i) Local access: When GUI and the application are on the same system, the administrator should provide security for the communications between the GUI and the application. The GUI must be given the least privilege and the application can be given higher privilege if required.

ii) Remote access: When the GUI controls the system remotely, the most important issue is how the GUI controls the application.

c) Web-Based Control:

A popular way to allow application administration is through a web interface, the advantage is that it does not require a dedicated client and can be used from multiple platforms.

3) Application Updates:

Keeping applications up to date with the latest security patches is one of the most important security measures.

Some mechanisms for easily updating applications are:

a) Manual updates

b) Automatic updates

c) Semi-automated updates

d) Physical updates

a) Manual Updates:

Manual updates require the administrator to physically download a file and install the update on the relevant system. This option is not preferred by the administrator as it consumes too much time.

b) Automatic Updates:

When an application uses automatic updates, it checks its website every so often for an update, and if one exists, it downloads it and installs it on the system.

There are two problems with this method:

i) Bandwidth usage

ii) Installing problematic patches

c) Semi-Automated Updates:

Some applications allow the administrator to decide when to download an update. After the update is downloaded, the application distributes the update to all the connected clients.

d) Physical Updates:

It's possible to update the system using an update received physically. In order to thwart an attacker from forging an update, the administrator can check for the size and CRC32 signature of the update at the vendor's site and compare it to the physical copy.

4) Integration with OS Security:

When an application is integrated with OS security, it can use the security information of the OS, and also modify it if required.

a) Importance of OS Security Integration:

OS security integration allows an application to either import or access in real time the OS's list of users and their privileges. The disadvantage would be if the administrator wants to enter the information of thousands of employees along with the information of their privileges, such a method would be time consuming and if the organization has more than one central system that requires manual user entry, this scenario would be even worse.

b) Manual Import of Security Information:

An application may allow the administrator to import all the user information and use it to manage authentication for the application. Although this method may speed up application deployment, there is still double administration afterward. For example, when an employee leaves the organization, the administrator has to delete the user both from the organization's user list and from the application list.

c) Automatic Integration of Security Information:

Automatic integration of security information allows the application to query the OS in real time for user credentials. This way, both the initial deployment time and the double administrative issues are solved.

There are two problems with this option, though:

i) If the OS's user database is deleted or lost, the application can't be accessed.

ii) The network connection between the application and the OS user database must be secured to prevent attackers from either eavesdropping on the line or using a fake server to gain information about users' credentials.

d) Using OS Security for Authorization:

An application can use OS security to authorize sessions. In this scenario, the application sets up a special directory or resource that can be accessed only by users who possess certain privileges, and the OS protects access to that directory or resource.

e) Keeping OS Security Integration Optional:

Sometimes it's necessary to deploy a small application that will be used by only one or two users, and if the application demands integration with the OS security in an organization with thousand users. Then it will only decrease the security and deployment speed.

Also the administrator may be reluctant to give an application the ability to modify (and potentially damage) the user directory.

11.10 REMOTE ADMINISTRATION SECURITY

Most of today's applications offer remote administration as part of their features, hence it must be secure, if an attacker manages to penetrate the administration facilities, other security measures can be compromised or bypassed.

11.10.1 Remote administration is needed for various reasons which are as follows:

1) Relocated servers:

An administrator needs an interface to administer any relocated web servers.

2) Outsourced services:

Managing security products requires knowledge that some organizations don't possess, so they often outsource their entire security management to a firm specializing in that area, the management of such security is done through the internet.

3) Physical distance:

An administrator may need to manage a large number of computers in the organization; these computers are physically separated by large distances. In such situations, it would become tedious and time-consuming to manage the system by physically attending, so remote access may make it simple.

11.10.2 Remote Administration Using a Web Interface:

Using a web interface to remotely administer an application or a computer has many advantages, but it also has its costs, and some advantages are also disadvantages.

Some advantages of remote web administration:

1) Quick development time:

Developing a web interface is faster than developing a GUI client, in terms of development, debugging, and deployment.

2) OS support:

A web interface can be accessed from all the major OSs by using a browser.

3) Accessibility:

A web interface can be accessed from any location on the Internet.

4) User learning curve:

An administrator knows how to use a browser, so the learning curve for the administrator will be shorter.

Remote web administration has some disadvantages, but they are usually not critical, they are as follows:

1) **Accessibility:** Because web administration is accessible from anywhere on the Internet, it's also accessible to an attacker who may try to hack it.

2) **Browser control:** Because a browser controls the interface, an attacker doesn't need to deploy a specific product control GUI.

3) Support Web-based applications are typically easier to support and maintain.

11.10.3 Authenticating Web-Based Remote Administration:

When connecting to the remote web administration interface, the first step is the authentication process. If the authentication is weak, an attacker can bypass it and take control of the application or computer.

HTTP Authentication Methods:

Some of the common methods to authenticate HTTP connections are

1) **Basic authentication:** When a page requires basic authentication, the user sends the encoded username and password using BASE64 encoding and sends it back to the server. If the login is correct, the server returns message number 200, which means everything is OK. If the login fails, it replies with the same 401 error as before.

2) **Digest authentication:** Digest authentication uses MD5 to hash the username and password, using a challenge supplied by the web server.

3) **Secure Sockets Layer (SSL):** SSL can be configured to require a client certificate (optional) and authenticate a user only if they have a known certificate.

4) **Encrypted basic authentication:** Basic authentication can be used in conjunction with regular SSL, thus encrypting the entire session, including the BASE64 encoded username and password.

5) **CAPTCHA method:** This is a popular method of verifying that the person on the other end is a human being, by showing a distorted image of letters and numbers and requiring the user to type them correctly.

11.10.4 Securing Web-Based Remote Administration:

The best solution for securely logging in to a web-administered server is to use either SSL, which checks for client certificates, or encrypted basic authentication.

Another option is to use secured custom logins (implemented with server-side scripts), but they may contain web exploits.

Custom Remote Administration:

Some applications are controlled remotely through a GUI or through console applications examples of such applications are SQL Server, Exchange Servers, firewalls, and intrusion detection systems (IDS).

Custom remote administration has both advantages and disadvantages.

The advantages of custom remote administration are:

1) **Complex graphics:** Sometimes the console needs to display complex graphics that can't be shown using a regular web administration interface.

2) **Authentication and encryption:** The application may use either a stronger authentication method or a stronger encryption method to secure the session.

3) **Availability:** Since the application can only be controlled from a dedicated GUI, the attacker will need to install it on his computer.

The disadvantages of custom remote administration are:

1) **Specific OS:** Some vendors will require a specific OS to run the controlling GUI, and the administrator will have to install it if it isn't already installed.

2) **Unavailability:** The application can be administered only from computers on which the GUI is installed, if the administrator is not in the office, it may not be possible to administer it from other computers.

11.10.5 Session Security:

It's important that the session between the client (GUI or console) and the application be secure. Otherwise, attackers may be able to gain information, steal credentials, or even conduct a replay attack. If the session is known to be insecure, the administrator can easily relay it through a VPN or a secure tunnel (SSH).

Authentication:

It's important that authentication takes place and that it should not depend on the IP or MAC address of the computer.

The sequence of the authentication process is also critical. The best way to exchange login information is either after the session is secured, or using a known method like EAP for insecure sessions.

Using OS Networking Services:

Some applications use OS networking services, such as remote procedure calls (RPC) or Distributed Component Object Model (DCOM), which allows the administrator to add data integrity, encryption, and authentication. If the OS security is non-trustworthy then either SSH or VPN connection can be used.

11.11 SUMMARY

Virtual machines present greater risks, because they provide computing environments that are based on software, which has inherent vulnerabilities, and because virtual machines are controlled by a master operating system known as the hypervisor. Attacks against vulnerabilities in the software that runs the guest operating systems or the hypervisor itself can lead to compromises in one, many, or all virtual systems in your infrastructure.

For that reason, special consideration must be given to the virtual environment. Securing the hypervisor is of paramount importance—it needs to be isolated from the guest OS and administrative access to it needs to be strictly controlled.

The guest OS themselves need to be protected with standard security software, as well as secure configurations within the virtual environment. Virtual storage and networks deserve the same consideration.

Cloud computing takes many forms, but they all have one thing in common—the Internet. And because cloud services are housed on the Internet, they carry all the risks inherent in the Internet as well as additional risks associated with the proximity of other users of the service, especially if any of those other users are malicious. Along with confidentiality risks that come from putting private data in the cloud, and integrity risks associated with the loss of direct control of data, the cloud also presents availability risks, because the Internet is an inherently unreliable medium. Real incidents that have been tracked by various agencies prove that service outage is the most commonly experienced security issue with commercial cloud services. Redundancy is the best way to mitigate those availability risks, just as with any other Internet service.

Application security needs to be done right from the start because it's much harder to actively fix security problems in the field than it is to do so in the programmer's chair.

Training, corporate standards, reviews at the design phase, and formal code reviews can all help ensure that security is integrated from the beginning in any new application.

Every programmer who isn't focused on security when writing an application, whether web-based or client can leave the application vulnerable to outside attackers. Because application security problems primarily result from human errors and omissions, the best solution is education.

To produce an application that is secure enough, define "secure enough" near the beginning of the development process. Keep this definition in mind when you construct each deliverable. As each deliverable is completed, check it for security issues. At the end of the development process, ship it only if the application meets your definition of secure enough.

11.12 QUESTIONS

1) What are Virtual Machines? What are the security requirements for a VM?

2) What are the ways to protect the hypervisor?

3) How is the guest OS protected in a Virtual environment?

4) What is Cloud computing? Give its types.

5) Elaborate on the benefits of Cloud computing.

6) Write a note on Risk and Remediation with respect to cloud computing.
7) What are the various cloud security technologies?
8) Explain the Confidentiality risks associated with cloud computing.
9) Explain the Integrity risks associated with cloud computing.
10) Explain the Availability risks associated with cloud computing.
11) Explain the secure development life cycle.
12) What are the Application security practices?
13) How is SQL injection performed?
14) What are the solutions to SQL injection attacks?
15) Explain the methods to exploit Forms and scripts.
16) What are the various ways for solving the data transfer problem?
17) Explain the following attacks

    a) Brute-force login b) Buffer overflows
18) Write a note on client application security
19) How remote administration is performed using a web interface?
20) Explain the HTTP authentication.

11.13 REFERENCES

- The Complete Reference: Information Security by Mark Rhodes - Ousley
- Essential Cyber security Science by Josiah Dykstra
- Principles of Computer Security: CompTIA Security+ and Beyond
- By Wm.Arthur Conklin, Greg White

*****

12

PHYSICAL SECURITY

Unit Structure

12.0 OBJECTIVES

The Objective of the chapter is to make the learner aware of various security breaches with respect to physical security and their solutions, the classification of assets and their security requirements for assets.

12.1 CLASSIFICATION OF ASSETS

Any resource having direct or indirect monetary value can be classified as an asset. An asset can be classified on the basis of criticality and the value each asset carries, based on it we need to develop security procedures and measures to protect them.

The classification of corporate physical assets will generally fall under the following categories:

1) **Computer equipment:** Servers, network-attached storage (NAS) and storage area networks (SAN), desktops, laptops, tablets, pads, etc.

2) **Communications equipment:** Routers, switches, firewalls, modems, private branch exchanges (PBX), fax machines, etc.

3) **Technical equipment:** Power supplies, uninterruptable power supplies (UPS), power conditioners, air conditioners, etc.

4) **Storage media:** Magnetic tapes, DAT, CD-ROM, Zip drives, hard drive arrays, solid-state drives, Secure Digital (SD), microSD, Compact Flash, and Memory Stick, etc.

5) **Furniture and fixtures:** Racks, enclosures, etc.

6) **Assets with direct monetary value:** Cash, jewellery, bonds, stocks, credit cards, personal data, cell phones, etc.

12.2 PHYSICAL VULNERABILITY ASSESSMENT

After classifying the asset, the physical security must be assessed. There are four main areas that must be part of physical vulnerability assessment.

1) Building

2) Computing devices and Peripherals

3) Documents

4) Records and Equipment

12.2.1 Building:

Physical Vulnerability assessment in this case can be done in the following ways:

1) Walking around the building to check for unlocked doors and windows.

2) Checking the areas around the building for obstructions such as bushes or shrubs.

3) Check for poor lighting conditions.

4) Also, check whether anyone can easily tailgate into the building.

5) Is it possible to walk into the building through an unattended gate?

6) Whether building passes are collected from the visitors once they leave the building?

12.2.2 Computing Devices and Peripherals:

Physical Vulnerability assessment in this case can be done in the following ways:

1) The accessibility and lockdown to systems and peripherals must be verified.

2) Unattended systems should be logged off or have their screens locked.

3) Critical Servers must be placed in a locked room, and access to the room must be through a card reader. (It can be used to audit.)

4) In case of a shortage of space logical isolation must be done.

5) The Physical lock must be used to protect the rooms and cases housing the resources.

6) BIOS must be password-protected through a complex password.

7) Booting from floppy/CD/DVD/USB drives must be disabled.

8) The monitor and the keyboard must be placed in such a way that they are only visible to the operator.

9) Unused modems and network ports must be removed or disabled.

10) Store tools separately, preferably locked up.

11) Limit the number of people with access to the server room, and document their access.

12) Place a sign-in sheet inside the door, or electronically track access with a card reader or biometric entry control.

12.2.3 Documents:

Physical Vulnerability assessment in this case can be done in the following ways:

1) The Documents must be classified as a part of data classification and information policies.

2) Confidential documents must not be lying around, a check must be made for them.

3) All the Post-it notes with passwords and credentials, documents not collected from print jobs and faxes, and documents in the trash or a recycle bin must be shredded.

4) A walk around must be carried to check if shoulder-surfing could be possible.

5) Employees must be educated on various espionage techniques.

12.2.4 Records and Equipment:

Physical Vulnerability assessment in this case can be done in the following ways:

1) Records generally contain employee timesheets, receipts, accounts payable/receivable, etc.

2) Records must be locked up when not in use and only be accessed by an authorized person.

3) Equipment items such as faxes, printers, modems, copiers, and other equipment have their own security recommendations, depending upon their use and location.

4) Devices such as smartphones or tablets must not be kept unlocked on the desk.

12.3 CHOOSING SITE LOCATION FOR SECURITY

When selecting a location for a data center or office site, survivability should be considered more important than cost. Selecting Low-cost sites may cause some unforeseen damages which could be more than the cost of selecting a high-cost site. If the selected site is in a flood zone or expecting a tornado or hurricane every year, or if it is seismic active or has a high crime rate then any one such event could cause a lot of expensive damage. Hence choosing a secure and reliable site location makes sense from a financial perspective as well as from a security point of view.

There are many security considerations for choosing a secure site location, some of which are as follows:

1) Accessibility

2) To the site

3) From the site (in the event of evacuation)

4) Lighting

5) Proximity to other buildings

6) Proximity to law enforcement and emergency response

7) RF and wireless transmission interception

8) Utility reliability

9) For a data center, the loss of power may be overcome through the use of generators, but if the water supply is cut off, the AC units will be unable to cool the servers.

10) Construction and excavation (past and present)

12.3.1 Accessibility:

1) Accessibility of the site is typically the first consideration and with a good reason.

2) If a site is remotely located then it would be difficult to commute, utilize and make practical usage.

3) If the site is easily accessible then it would be accessible for others too (maybe attackers).

4) The site should be such that, in case of any untoward incident such as a bomb threat, fires, or terrorist attacks, the evacuation must be easily carried out.

12.3.2 Lighting:

1) Proper lighting, especially for organizations with 24×7 operations, should be evaluated and taken into consideration.

2) Poor lighting results in threats to employee safety and is a potential for break-ins.

3) Mirrored windows or windows with highly reflective coatings should face north-south rather than east-west.

4) Lighting should be positioned in such a way that it never blinds those leaving the building at night.

12.3.3 Proximity to Other Buildings:

1) Sharing a building with a branch of law enforcement would be considered less risky than sharing a building with pubs and clubs.

2) The closer the proximity to other buildings and companies, the higher the probability is for a physical security incident to occur.

3) Any problems in an adjacent or connected building might have could potentially become our problem as well.

12.3.4 Proximity to Law Enforcement and Emergency Response:

1) The location of organizations relative proximity to law enforcement and/or emergency response units can be useful.

2) In areas having a history of crime, delays to get a response from the agencies could cause significant loss.

3) If an emergency service unit were to be called to respond to an incident and there was a delay in the response, then it would also cause dearly to the organization.

12.3.5 RF and Wireless Transmission Interception:

1) With wireless networking becoming more prevalent, wireless hacking and hijacking become more of a threat.

2) Wireless protocols such as radio frequency devices, cordless phones, cell phones, PIMs, and mobile e-mail devices must be taken into consideration.

3) Scanners must be used to check the vulnerability of the protocols.

4) Frequency ranges that are heavily used must be avoided.

5) Encryption must be used for sensitive traffic.

12.3.6 Utilities Reliability:

1) Problems such as power outages, network outages, and disruption in phone services can cause serious damage to the organization.

2) Power outages can be compensated by using UPS systems or generators (both having their own shortcomings).

3) Network issues and phone service disruption can be taken care by switching to other operators, but this is not possible always.

4) While replacing old wires with new ones can also cause downtime.

5) For data centers loss of power can make a serious impact, as it requires constant cooling.

12.3.7 Construction and Excavation:

1) Construction and excavation can take the entire network and communications infrastructure down in one go.

2) Past construction activities in the area, must be noted.

3) Town or city records provide information regarding any construction/excavation/demolition, both past, and present.

4) Information about power/telecom outages must be obtained.

12.4 SECURING ASSETS: LOCKS AND ENTRY CONTROLS

Some more methods in securing physical assets.

12.4.1 Locks:

1) Any asset used in the organization must be secured in a location with a lock.
2) Devices such as laptops, smartphones, tablets, MP3 players, jewelry, and keys, must be secured using locks.
3) Asset owners must be educated about its importance.

12.4.2 Doors and File Cabinets:

1) Locked doors must be checked.
2) The functioning of the door must be checked properly.
3) Doors must withstand sufficient force.
4) File cabinets containing sensitive information or valuable equipment should be kept locked when not in use.
5) The keys to these should also be kept out of common reach.

12.4.3 Laptops:

1) Laptops at the office should be physically locked to the desk.
2) Cable locks ensure that the laptop doesn't fall into the wrong hands.
3) Laptop theft has become most common, so special care must be taken while travelling.
4) While going through a metal detector special care must be taken.
5) Operating system security and software safeguards must also be considered.

12.4.4 Data Centers, Wiring Closets, Network Rooms:

1. All of these areas should have common access controls since they all perform a similar function.
2. Rooms housing these resources must be kept locked.
3. If automatic entry-tracking mechanisms are not in use then the access log must be kept.

12.4.5 Entry Controls:

1. Entry controls have their own security considerations which change with the security plan and business needs.

2. It is first necessary to locate the site where the entry controls need to be deployed.

3. Some of the common scenarios are an existing structure with a single tenant, a suite in a multitenant building, a campus group of buildings with specific public entrances, and a high-rise building.

12.4.6 Building Access Control Systems:

1. Many existing structures may already have an access control system, which can be reused.

2. Multitenant buildings typically have access control systems that control entrance into the building or entrance to a special parking area which is common to the entire building.

3. In order to implement an access control system that is not compatible with an existing system, multiple access cards may be necessary.

4. The most important factor when dealing with a multitenant building is to make sure that anyone passing from the unsecured region to the secured region must pass through the security check.

12.4.7 Mantraps:

1. A mantrap is an area designed to allow only one authorized individual entrance at any given time and prevent an unauthorized person from closely following an authorized person through an open door (an anti-tailgating mechanism).

2. Typical areas of concern are high-security areas, cash-handling areas, and data centers.

12.4.8 Building and Employee IDs:

1. Any organization hiring new employees must provide them with ID badges.

2. Building and/or employee identification should be displayed at all times.

3. Anyone not having a visible ID should be challenged.

12.4.9 Biometrics:

1. A biometric device is classified as any device that uses distinctive personally identifiable characteristics or unique physical traits to positively identify an individual.

2. Some of the most common devices employing biometrics follow the characteristics such as fingerprint, voice, face, retina, iris, handwriting, hand geometry, and keystroke dynamics. The most commonly deployed biometric technologies are currently fingerprint and hand geometry devices.

3. The latest fingerprint readers now read the corpuscles under the skin, so they can be used for nearly everyone, even individuals who do not have strong fingerprint ridges.

4. The recent trend of implementing fingerprint readers in commercial devices such as laptops and time and attendance devices has resulted in this technology becoming more cost effective.

12.4.10 Security Guards:

1. A security guard is not just a person but also a resource.

2. Security Guards are the best deterrent.

3. They also perform the duties such as patrolling, guarding, monitoring, preserving, protecting, supporting, and maintaining the security and safety of personnel and property.

4. Security guards deter, detect, and report infractions of organizational rules, policies, and procedures.

5. Security guards help limit or prevent unauthorized activities, including but not limited to trespass, forcible entry or intrusion, vandalism, pilferage, theft, arson, abuse, and/or assault.

6. Guard placement, number of guards, and use would be as per the requirement.

7. Background checks should be done for all security guards, and appropriate licenses and clearances obtained wherever applicable.

12.5 PHYSICAL INTRUSION DETECTION

Physical intrusion detection requires forethought, planning, and tuning to obtain optimal effectiveness. Some security considerations for physical intrusion detection are as follows

12.5.1 Closed-Circuit Television (CCTV):

1. CCTV must be placed considering the financial and operational limitations of the organization.

2. They must be placed to cover high-traffic areas, critical function areas (such as parking structures, loading docks, and research areas), cash-handling areas, and areas of transition.

3. The cabling used for CCTV devices must not be readily accessible, making tapping difficult.

4. Lighting will also play a critical role in the effectiveness of the camera.

5. When installing wireless CCTV, a transmission must be checked for any interception by an attacker.

12.5.2 Alarms:

1. Alarms should be tested every month and a test log must be maintained.
2. Points of entry and exit should be fitted with intrusion alarms.
3. A response plan should be ready in advance in case of any intrusion.
4. Duress alarms (silent alarms used in the time of distress) should also be taken into consideration for areas that may require them.

12.6 SUMMARY

There are many physical security considerations that should coincide with your data security goals. Both physical and data security are centered on the protection of assets, so some concepts apply directly to both worlds. Common sense, forethought, experience, and clear, and logical thinking are an essential part of any security plan.

12.7 QUESTIONS

1) What is an asset? Give the classification of Assets.
2) Explain Physical Vulnerability assessment.
3) What factors must be considered while selecting a site location for security?
4) What are the ways to secure assets using physical security devices?
5) What role do security guards play in providing security?
6) What are the ways to detect and alert any physical intrusion?

12.8 REFERENCES

- The Complete Reference: Information Security by Mark Rhodes – Ousley Essential Cyber security Science by Josiah Dykstra
- Principles of Computer Security: CompTIA Security+ and Beyond By Wm.Arthur Conklin, Greg White

*****